UDC 004.78:658.012.4

RESEARCH ON LIGHTWEIGHT REAL-TIME COLLABORATIVE MANAGEMENT SYSTEM FOR ERP OF SMALL AND MEDIUM-SIZED ENTERPRISES BASED ON NODE.JS

Zheng Xiaohui, master's degree student,
Sednina M. A., senior lecturer
Mihutski I. E., PhD in Technical
Belarusian National Technical University
Minsk, Belarus

Abstract. Against the backdrop of SMEs' digital transformation, traditional ERP systems face pain points like high deployment costs, long customization cycles and insufficient real-time collaboration. This thesis develops a lightweight real-time collaborative erp system following the "theory-design-practice" framework. It clarifies contradictions between traditional ERP and SMEs' demands, constructs relevant theories and methodologies, designs a four-tier architecture and core modules, and develops the system using Node.js (NestJS), Vue3, MySQL and Redis. Multi-dimensional testing and 3-month pilot applications in two SMEs verify its feasibility. Core innovations lie in integrating lightweight design and WebSocket-based real-time collaboration, providing a low-threshold solution for SMEs' digital transformation.

Keywords: Small and Medium-Sized Enterprises (SMEs); ERP Management System; Lightweight Design; Real-Time Collaboration; Micro-Frontend; Digital Transformation.

1. The Theoretical and Methodological Foundations of the Lightweight Real-Time Collaborative ERP Management System for Small and Medium-Sized Enterprises.

1.1. Relevant Theoretical Foundations.

This section sorts out ERP's evolution from MRP to modern integrated systems. Taking typical SMEs as samples, it analyzes traditional ERP's core dilemmas: high cost, long cycle and low flexibility [4, 7]. It clarifies lightweight ERP's three characteristics: modularization, multi-mode deployment and intelligent interaction, and introduces real-time collaborative management theories to support subsequent design.

1.2. Key Technical Methodology System.

This section focuses on five core technical fields and compares mainstream technology stacks. It determines optimal combinations for SMEs: Node.js (NestJS) for back-end and qiankun for micro-frontend [1, 5]. Key technical solutions include "local+Redis cache" for delay reduction [8, 13], RESTful API with rate limiting [11], ensuring system stability.

1.3. System Research Framework and Methodology.

This section establishes a "theory-technology-practice" three-tier framework and a lightweight technology selection index system. It designs a "permission-process-conflict" collaborative model and refines two lightweight principles: module splitting and on-demand resource allocation, ensuring research scientificity.

This chapter clarifies why SMEs need lightweight ERP, optimal technical paths and system model construction, filling theoretical gaps and laying foundation for subsequent design and implementation.

2. The Design of the Lightweight Real-Time Collaborative ERP Management System for SMEs.

2.1. Overall Architecture Design.

This section designs a four-tier architecture: front-end presentation (Vue3), API gateway, business service (microservices) and data storage (MySQL + Redis) [1, 9, 13]. A multi-mode deployment scheme (SaaS, private cloud, local) is proposed to adapt to different SMEs' IT conditions [14].

2.2. Core Module Design.

Centering on SMEs' core businesses, four modules are designed: order management (full-process control), inventory collaboration (real-time monitoring and early warning), financial connection (automatic voucher generation) and user authority (RBAC model). Inter-module data linkage realizes "order-inventory-finance" collaboration.

2.3. Key Technology Design.

Key technical schemes: qiankun for micro-frontend integration [5]; NestJS Gateway + Socket.io for real-time communication [16]; cache optimization to solve penetration/breakdown/avalanche issues [13]; JWT+RBAC for security protection [12], with API and data transmission encryption.

2.4. Non-Functional Requirement Design.

Non-functional requirements: operation response time ≤ 3 s, supporting ≥ 50 concurrent users; data security with encryption and operation logs; user-friendly interface with 1-day learning cost, meeting SMEs' practical needs.

This chapter transforms theories into practical design schemes, meeting SMEs' lightweight deployment and real-time collaboration needs, providing technical guidance for subsequent development and verification.

3. The Implementation and Verification of the Lightweight Real-Time Collaborative ERP Management System for SMEs.

3.1. System Development and Implementation.

Using agile development, the system is built with Node.js (NestJS), Vue3, MySQL and Redis, and deployed in three modes. Core technologies like real-time communication and cache optimization are implemented and debugged to ensure stable operation [14, 16].

3.2. System Testing.

Functional, performance, compatibility and security tests are conducted via simulated scenarios. Results confirm module integrity and system stability [12].

3.3. Real-Scenario Verification.

3-month pilot in two SMEs shows: manufacturing enterprise's order efficiency up 35 % and inventory overstock down 22 %; retail chain's settlement cycle shortened from 5 to 2 days [7, 14], verifying efficiency improvement effects.

This chapter verifies the system's value and the validity of previous theories and designs, completing the research loop and providing a reusable solution for SMEs.

CONCLUSIONS.

Targeting traditional ERP's pain points in SMEs, this thesis conducts “theory-design-practice” research on lightweight real-time collaborative ERP. Core conclusions are as follows.

Achievements: Established lightweight real-time collaborative ERP theory, designed four-tier architecture with multi-mode deployment, and developed/verified the system, which is proven low-cost, flexible and collaborative for SMEs.

Innovations: Integrated lightweight design (module splitting, on-demand allocation) and built real-time collaborative mechanism, solving SMEs' collaboration issues.

Limitations & Future Work: Limitations include narrow function coverage and single verification samples. Future work will expand functions, integrate AI [14], broaden verification scope and explore IoT/blockchain integration [15].

Table of contents

1. Chaplia, O. Enhancing REST API Handlers Organization for Node.js Microservices / O. Chaplia [et al.] // SN Computer Science. – 2025. – Vol. 6, № 7.

2. Tigrine, A. A web-based system for real-time ECG monitoring using MySQL database and DigiMesh technology: design and implementation / A. Tigrine [et al.] // Medical & Biological Engineering & Computing. – 2025. – (Prepublish).

3. Zhou, Q. Research on the Design and Implementation of Case Teaching in MySQL Database Course / Q. Zhou, J. Zhang // Education Reform and Development. – 2025. – Vol. 7, № 5.

4. Sulaiman, A. Role of ERP systems in management accounting in SMEs in Saudi Arabia / A. Sulaiman // Journal of Accounting & Organizational Change. – 2025. – Vol. 21, № 2.

5. Rafi, M. Implementing Micro Front End Architecture: Vue.Js' Diversified Strategy / M. Rafi // Journal of Research in Science and Engineering. – 2025. – Vol. 7, № 2.

6. Aparecido, A. R. Identifying the maturity level of ERP systems implementation and use in SMEs: a study using grey fixed weight clustering / A. R. Aparecido [et al.] // Grey Systems: Theory and Application. – 2025. – Vol. 15, № 1.

7. Salunke, V. S. A Performance Benchmark for the PostgreSQL and MySQL Databases / V. S. Salunke, A. Ouda // Future Internet. – 2024. – Vol. 16, № 10.

8. Sravani, C. Constructing a Study Buddy Using MERN (MongoDB, Express.js, React, Node.js) Stack Technologies / C. Sravani [et al.] // Engineering Proceedings. – 2024. – Vol. 66, № 1.

9. Liu, Y. Campus Second-Hand Textbook Trading Platform based on Vue 3 and Spring Boot / Y. Liu // International Core Journal of Engineering. – 2024. – Vol. 10, № 6.

10. Tyson, M. Intro to Nest.js: The higher-order JavaScript and TypeScript server / M. Tyson // InfoWorld.com. – 2023. – URL: www.infoworld.com (date of access: 16.10.2025).

11. Kanjilal, J. How to use OpenAPI in ASP.NET Core / J. Kanjilal // InfoWorld.com. – 2023. – URL: www.infoworld.com (date of access: 16.10.2025).

12. Ksenia, P. CSRF protection in JavaScript frameworks and the security of JavaScript applications / P. Ksenia, C. Xiuzhen // High-Confidence Computing. – 2021. – Vol. 1, № 2.

13. DigitalOcean Unveils Managed Databases for MySQL and Redis // Wireless News. – 2019.

14. Yong, W. Research and Design of ERP System for Small and Medium-sized Enterprises under Great Intelligence Mobile Cloud / W. Yong // IOP Conference Series: Materials Science and Engineering. – 2019. – Vol. 646.

15. Cahyadi, I. A Combined ANP, TOPSIS and MCGP Approach to Select Knowledge Transfer Strategy: A Case Study in Indonesian SMEs ERP System Implementation / I. Cahyadi // IOP Conference Series: Materials Science and Engineering. – 2019. – Vol. 505, № 1.

16. Setyawan, S. Sistem Push Notification Life@BankJateng Sebagai Penunjang Layanan Pegawai PT. Bank Jawa Tengah Menggunakan Socket.IO / S. Setyawan, A. Susanto // JOINS (Journal of Information System). – 2018. – Vol. 3, № 2.

UDC 658.152

RESEARCH ON THE IMPORTANCE OF ZTO EXPRESS'S FINANCING STRUCTURE TO ITS OWN DEVELOPMENT

Han Jiali, master's degree student
Scientific supervisor – PhD in Technical Sciences, associate professor Yarmolchick Yu. P.
Belarusian National Technical University
Minsk, Belarus

Abstract. This study explores how ZTO Express's post-listing financing structure affects its growth, using its 2016–2024 data and empirical methods.

Keywords: ZTO, Express Financing, Structure Corporate, Growth, Post-listing, Empirical Methods.

1. Introduction.

Background: China's logistics grows via e-commerce/digitalization; ZTO listed in NYSE (2016) faces competition, cost pressures, and needs for capital to expand/upgrade, making financing structure critical.