

# Design and Implementation of Online Mall System based on Java Web

Zengyu Cai<sup>a</sup>, Yuanbo Liu<sup>a</sup>, Yong Gan<sup>b,\*</sup>, Jingxiao Li<sup>a</sup>, and Yuan Feng<sup>a</sup>

<sup>a</sup>*School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, 450002, China*

<sup>b</sup>*Zhengzhou Institute of Engineering and Technology, Zhengzhou, 450044, China*

---

## Abstract

In regard to e-commerce, this paper adopts the current mainstream framework technology and distribution architecture and proposes a high-performance online mall design. Firstly, the service architecture of the online mall system is introduced in detail. Then, the main functional modules of the system are described, and the business process of the system is analyzed. Finally, the key technologies used in the system are presented. The relevant design schemes proposed in this paper have important reference value for small and medium-sized enterprises to build low-rise stations, and the solution can also guide developers.

*Keywords:* online mall; system performance; web; system architecture; Mysql

(Submitted on September 16, 2019; Revised on October 11, 2019; Accepted on November 25, 2019)

© 2019 Totem Publisher, Inc. All rights reserved.

---

## 1. Introduction

With the development of the Internet, e-commerce has also become an indispensable part of today's society. E-commerce is a very convenient marketing model [1]. It is not only limited to major e-commerce platforms; many small and medium-sized enterprises can still rely on self-demand to establish unique e-commerce platforms. However, e-commerce has high development and operating costs [2]. Therefore, through the analysis and examination of the e-commerce model, this paper presents a high-performance and low-input cost-effective mall website development design and analyzes the mainstream technology, payment methods, and distributed services used in the system.

## 2. Web-based Online Mall Design

### 2.1. Master Plan

Before starting the development of e-commerce websites, it is necessary to analyze the relevant business needs. Therefore, the most common principal line business logic for the regular online shopping mall system is to select commodities, pay, and obtain the purchase of goods successfully.

There are many other business needs, such as user reviews, returning goods, and so on. In the design of the system, there are requirements of the online mall related to the system design, such as the B2C mode e-commerce website [3-4]. In the system foreground design, the login user only needs to operate on the goods, such as the user browsing the goods, the user handling the shopping cart, the user paying, order management, user comments, and other functions. In the product back-end management, the administrator has absolute management authority over the mall website, thus requiring user management, product management, overall page management, comment management, order management, and other operations to complete order processing. Website maintenance ensures the integrity of the website's functionality.

### 2.2. System Architecture Design

The online mall system is based on Java web development. The data B/S structure is easy to operate. In system development,

---

\* Corresponding author.

E-mail address: [zzulilyb@163.com](mailto:zzulilyb@163.com)

we use the relatively mature Spring+SpringMVC+Mybatis-Plus framework. Compared with the old Spring +Stratus2 +Hubitan framework, it has strong security and better SQL optimization support. Mybatis-Plus is light and easy to use, and it has anti-SQL injection and an integrated back-end paging plug-in. Thus, it is more convenient and safe to use [5]. In the system architecture, the system development uses distributed service deployment, as shown in Figure 1.

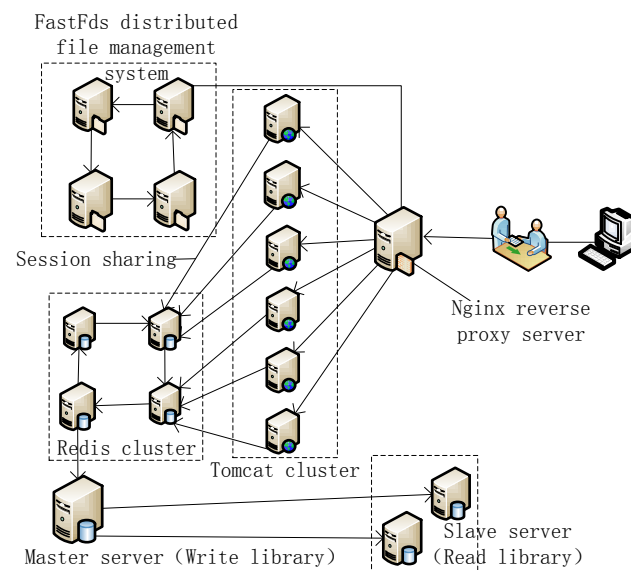


Figure 1. System architecture diagram

**Nginx proxy server:** Nginx is a high-performance reverse proxy server [6] that can proxy web server clusters, forward user requests through configuration weights, and realize data and permission sharing between servers through Redis storage session. Clustering achieves the horizontal expansion of server hardware and can prevent single server failures from leading to downtime of the entire service.

**FastFds distributed file management system:** FastFds has the advantages of speed, a large capacity, and ease of expansion [7]. This system is mainly used to store a large amount of product image information in a page and reduce the user request page when a large amount of data load is caused by the bandwidth footprint.

**Redis distributed cluster:** In the mall website, a major problem is the frequent loading of the mall data. When the number of users is very large, the face is not just the page data that is frequently loaded. Caching is important to reduce the pressure of the database, which will achieve the goal of system performance improvement [8-9].

**Mysql read and write separation:** The online database system faces the most database operations as reading operations. Therefore, in the system architecture, the MYCAT middleware is used to read and write the database to improve the overall performance of the database.

### 2.3. System Function Design

In the traditional shopping mall website, the purpose of the user visiting the network station is to purchase his or her favorite goods. Therefore, in the online shopping mall design, there should be functions such as product display, product classification, shopping cart, order, and payment. In order to complete all the processes of online shopping, the function of the online mall system is divided roughly into six modules. The system module is shown in Figure 2.

Through the reasonable interaction between the various functional modules of the system, the integrity of the online mall is realized. The user can simplify the offline shopping process through the operation of each module [10], thereby improving the system operating efficiency.

**Carousel map module:** The administrator can prompt the user to the latest information or the newest product through the carousel module. The user can view and click to view the details on the home page. In background management, administrators can sort rotations, add links, disable carousels, and replace carousels.

User module: In this module, the user can register or log in. When the user registers, he or she must fill in the registration information form completely. The registered account needs to be activated by email authentication. If the user is not activated, he or she cannot log in to the system. The comment module, shopping cart module, order module, and payment module are operated.

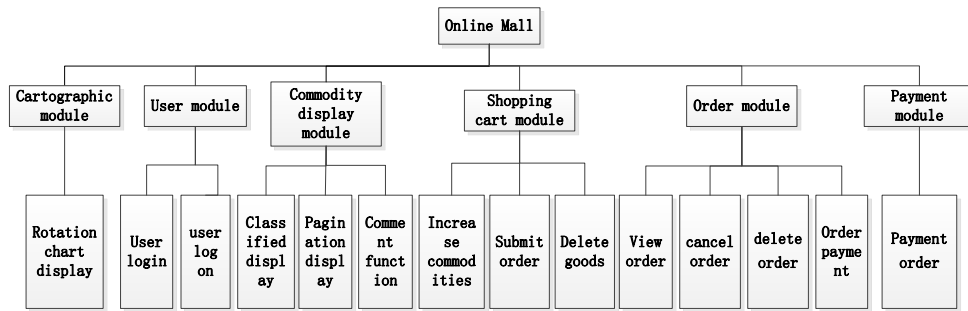


Figure 2. System function module diagram

Product display module: This part is mainly for the display of product information. In this module, there is a classified navigation display, web presentation, and user comment function. Through the back end, the user searches for the item of interest through the classified navigation bar on the home page. After entering the product details, the user can comment on the product and view the product review after logging in.

Shopping cart module: After the user adds the product to the shopping cart, the user can add and delete the goods in the shopping cart, determine the order after purchase, automatically clear the shopping cart, and jump to the order management module.

Order management module: Users can operate on existing orders, such as canceling orders, viewing orders, deleting orders, and paying for orders. When the user clicks on the payment order, the payment module appears.

Payment module: Considering the payment security problem, the system adopts the Alipay interface, which is currently widely used. The user automatically calls the Alipay interface after clicking the payment. The user can pay through the Alipay client or log in to the Alipay page through the Alipay account. The payment is completed.

## 2.4. Database Design

To improve the expansibility of the system, the coupling between the tables is reduced by avoiding the operation of linking tables in the database design. After many considerations, MySQL is used to design the table of the system, and we add the index to the relevant retrieval fields of the data table to improve the retrieval speed.

In this specific implementation, we show the overall database table relationship of the system in Figure 3.

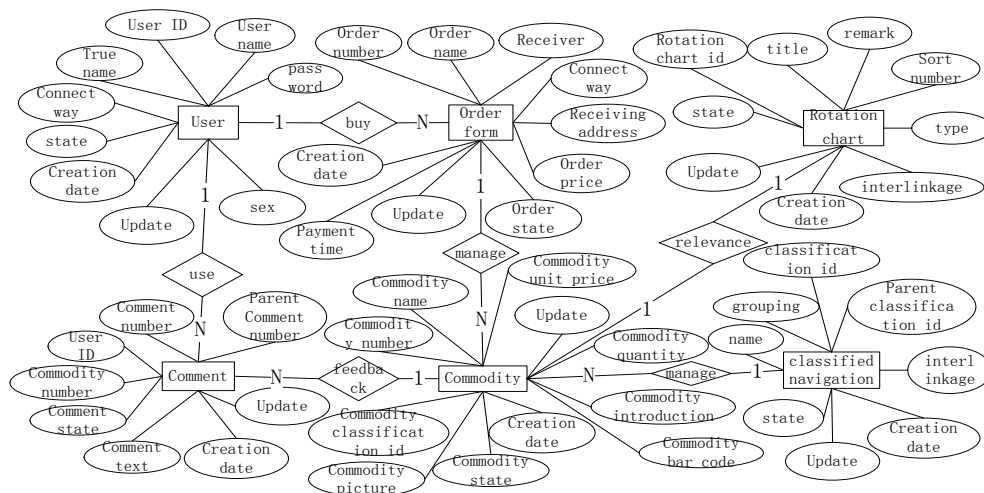


Figure 3. ER diagram

**User table:** In the user table, it is necessary to record the user login name, user login password, user gender, real user name, user contact information, user account status, and other information in order to ensure the uniqueness of the user in the system. The user number is the primary key, and the user name cannot be the same. In the table relationship, the user table has a one-to-many relationship with the order table; that is, multiple orders can exist under the same user. The user table also has a one-to-many relationship with the comment table; that is, multiple comments can be present under the same user.

**Order form:** The order form is mainly used to store order information in the order form, including the order name, consignee information, and order status. In the case of user login, the system finds and displays the corresponding order list through the user ID. In the table relationship, the order table has a one-to-many relationship with the item table; that is, there may be multiple items in one order.

**Rotation chart:** The rotation chart is mainly used to store the information of the rotation chart on the home page, including the carousel map ID, carousel title, carousel front-end display sorting number, and carousel jump link. In the table relationship, the carousel map has a one-to-one relationship with the product table; that is, one carousel map corresponds to the product detail.

**Comment chart:** In the product details, the user can comment on the product after logging in, so we need to store the user number, product number, parent comment number, comment content, comment status, and other information in the comment table. In this system, the system status is used to determine whether to display the comment to prevent inappropriate comments. In the table relationship, the comment table has a one-to-one relationship with the product; that is, one comment corresponds to one item.

**Product list:** The product table is mainly used to store product-related information, such as product name, product quantity, product price, product classification ID, product image link address, and other information. In the system relationship, the product table has a many-to-one relationship with the order table, the product table has a one-to-one relationship with the carousel chart, the product table has a one-to-many relationship with the comment table, and the product-to-category navigation table has a many-to-many relationship.

**Classification navigation table:** The classification navigation table is mainly used to store commodity classification information in the classification navigation table. In the online mall design, we divide the classification navigation into groups, and when the parent classification ID is 0, the node is the root node. In the system relationship, the classified navigation table has a many-to-one relationship with the commodity table; that is, there may be multiple commodities under one category.

### **3. Key Technologies of Online Mall System**

#### *3.1. Redis Cache*

In the system, the page data is cached mainly through Redis. Also, it is more important in the website of the electronic mall category, and it is more difficult to solve the inventory problem of the product [11]. Therefore, in this system, the quantity of the goods is stored in the cache through the set command of the Redis, and the destock operation is carried out by using the Decrby directive. If the return value is less than zero, the purchase amount of the product is insufficient, and we cannot execute the purchase operation. At the same time, we use the Incrby command to increase the inventory operation, because the Redis operation is atomic. Thus, when adding or subtracting inventory, add a distributed lock to the operation to prevent "dirty reading." This method reduces the number of database reads and writes and improves the overall performance of the system.

Secondly, in the process of processing the user to submit the order, the same cache is used, so the user presents the order into the cache and sets the timeout period [12]. If the order times out, the order is stored in the database, and the order status is set to timeout. Orders will no longer be able to make payment operations. When the user pays the order, the order is also put into the database, and the order status is set to paid. We store the rest of the operation orders in the cache, and the database is not operated. This approach can reduce database operations.

#### *3.2. Safety Design*

The most indispensable thing in the e-commerce industry is system security. In this system, the transaction of funds is involved. Therefore, we cannot underestimate the safety problem. Secondly, the user information security problem requires each user to log in through the account during the entire shopping process. The operation and its security are equally important. We implement the security of this system in the following three aspects:

**Payment safety:** In the development of this system, we do not adopt the user account balance payment, because the introduction of user balance will increase the risk of user account theft and the loss of user funds, so the system uses the Alipay third-party payment tool and calls the Alipay interface. The merchant UID is bound, and the message is verified through RSA2 to ensure the correctness of the user's payment request and avoid the occurrence of chain theft [13]. Alipay has a perfect operating mechanism. Users can view through the Alipay mobile phone client and do not need to worry about payment after the payment is completed, so the system has strong payment security.

**Data safety:** This system adopts the more mature framework, which can avoid many database security problems. At the same time, the spring framework is also compatible with many open source security frameworks, such as the Shiro security permission control framework. We also set up the MySQL database [14]. We connect the proprietary administrator password by verifying the password when connecting. The front and back windows in the system are separated, so the background management system can be independently deployed in the internal network to ensure that the background management system is not subject to the external network attack.

**Web server safety:** In the system deployment, we mainly use the Tomcat server, so the Tomcat server is configured to accept only the requests sent by the specified IP. We specify IP as the Nginx proxy server, and then the Nginx reverse proxy is used to achieve the purpose of other IP accessing Tomcat services [15]. This way can protect the Nginx server by configuring the firewall, configuring the Nginx security policy, adding the necessary security services, and so on to achieve the security operation of the Nginx service.

## 4. Performance Optimization of Online Mall System

### 4.1. Web Front End Optimization

The implementation of the online mall system architecture mainly uses distributed Redis caching technology, clustering technology, and distributed file systems. After the analysis of the system architecture design above, the key nodes are displayed through the corresponding optimization techniques.

(1) Reduce static resource loading and front-end page statics. Reduce HTTP requests as much as possible in front-end optimization, that is, reduce the introduction of js and other static resources. Secondly, through the test lookup, the request timeout problem caused by the static resource 404 and the front-end irrelevant CSS style is reduced, and the unnecessary and repeated js scripts are deleted. Image compression reduces the number of HTTP requests and request traffic. We use page static processing technology, that is, statically process pages that are not updated for a long time, and delete static pages when updating data. If the static page exists when the user requests it, it will be accessed directly; otherwise, the static page will be regenerated.

(2) Use front-end framework and back-end paging technology. The spring framework in the online mall system is fully compatible with the Velocity template language, so the language is used instead of the traditional JSP page. At the same time, we use a more mature bootstrap front-end framework to replace the system's native page, making the front-end code simple and avoiding too much duplicate code. The system uses back-end paging instead of front-end paging and uses the paging plug-in and asynchronous request to pass the page number and page size to the back-end to obtain data, effectively preventing the load delay caused by large data volume. The jquery method is used to dynamically refresh the data to reduce the number of page HTTP requests.

### 4.2. Database Optimization

Compared to the optimization of the front-end page, the performance optimization of the database is more troublesome. First, the system requirement analysis should consider the horizontal expansion of the database system, so the database needs to avoid the joint table query to reduce the coupling between the database tables.

(1) Use caching to avoid frequent reading and writing of data. In Redis, we use the counter to determine whether the user has participated in the snap; if not, the counter is incremented by one and the user snapped information is written to the cache. When the counter is 50, all subsequent snap requests are rejected, thereby achieving the purpose of stream clipping. However, using this kind of reading and writing in this case is very dangerous, so we use the principle of snapping concurrency to reduce the number of database reads and writes.

(2) Optimize a plan for frequent order operations. We consider that the user's operation is irregular; for example, the user adds the commodity and generates the order-locked item, but the long-term unpaid amount causes the inventory to be

occupied. In this case, the user does not directly operate the database when generating the order but stores the user's operation and order information in the Redis cache without persisting it. After the user completes the payment, the order information is inserted into the database, and the order information is cleared in Redis. This not only improves the speed at which users submit and view orders, but also reduces the pressure on the database caused by frequent reads and writes.

(3) Consider other optimization methods. In addition to the above caching mechanism, performance improvement requirements are also met by increasing data caching, query statement optimization, avoiding large-scale join table queries, increasing the database table index, back-end page queries, and database table storage.

#### 4.3. Server Optimization

In order to reduce the server system pressure of page access requests and ensure that the system will not crash due to image problems, we have different configuration optimization methods on the Tomcat application server and Nginx server.

Tomcat's default configurations of memory and threads are relatively low, which will effectively reduce server performance and concurrency, so Tomcat optimization mainly considers four directions: Tomcat memory optimization, Tomcat thread optimization, Tomcat IO optimization, and session storage optimization. This not only reduces server memory consumption, but also makes the code simple and readable, which is more conducive to later optimization.

The optimization of the Nginx server is mainly reflected in the number of configuration service processes, the number of process connections, the number of files opened simultaneously by the system, and the opening of the sendfile. Finding the best configuration within the server's memory allowable range can greatly improve system responsiveness and effectively reduce server load pressure.

### 5. Online Mall System Function Test

#### 5.1. Functional Testing

The online mall system based on Java web mainly has six functional modules, which are a carousel module, a user module, a product display module, a shopping cart module, an order management module, and a payment module. This system was developed based on the Java platform and runs on the Windows platform.

In the aspect of function test, we test the various functions of the online shopping mall system based on Java web according to system design. We show the test results in Table 1. The test results show that we realized all the functions involved in this system: to meet the requirements of development function design and achieve the browsing, management, and maintenance of goods. The test results show that the system has realized the function of the online shopping mall system and has a better application prospect. It has laid the foundation for the next step of improvement.

Table 1. Functional test results

Ownership module	Name (of a thing or organization)	Function	Status
Order module	View an order	View the details of the order by selecting	Realize
	Cancel an order	Cancel existing orders	Realize
	Delete an order	Delete management of existing orders	Realize
	Pay for an order	Payment of orders that exist in the mall	Realize
Shopping cart module	Increase commodities	Add new items to the shopping cart	Realize
	Delete goods	Delete items that already exist in the shopping cart	Realize
	Submit an order	Submit the current order	Realize
Commodity display module	Classified display	Classify and display existing commodities	Realize
	Pagination display	Pagination shows existing goods	Realize
	Comment function	Ability to comment on goods	Realize
User module	User login	The user can successfully log in to the mall	Realize
	User logs on	Users can register	Realize
Cartographic module	Rotation chart display	Take turns to display goods	Realize
Payment module	Payment order	Payment of order	Realize

#### 5.2. Performance Optimization Test

In order to confirm that the above work does improve the performance of the online mall system compared with the traditional system, we used the stress-testing tool Siege in this test. The tool can enable multi-user concurrent access to websites and records the response time of each user request, which can be used to simulate user request loads and

implement stress testing. Through the Siege performance testing tool, Table 2 shows the analysis of test results. The test is optimized on the same machine before and after optimization, and the network environment is consistent, so the test has high credibility.

Through the comparison of the above test data, it can be seen that the average performance of the system has been greatly improved after the performance optimization of the system. Due to insufficient computer condition resources and lower test concurrency, the test gap is small. However, the optimized system has strong scalability and can fully support the concurrent needs of small and medium-sized websites. The optimization method mentioned can also be effective if it can be flexibly applied in small and medium-sized websites.

Table 2. Comparison data of system stress test (multiple experimental results)

Project	Before optimization	After optimization	Remark
Total number of completed transactions (hits)	20629	22859	Total number of transactions processed by the system during the test case
The success rate of completing the transaction (%)	96.0-100.0	97.0-100.00	Success rate of the system processing test case transactions
Total time (sec)	170.0-190.0	100.0-120.0	Total time used to complete the stress test
Total response data size (MB)	1250.60	1269.23	System test case request always wants data size
Display network connection speed (sec)	0.25-0.8	0.45-0.8	Current network latency in system testing
The average number of transactions per second (trans/sec)	105.0-145.0	181.0-192.0	Average number of request transactions processed per second by the system
The average amount of data transferred per second (MB/sec)	5.0-10.0	5.0-10.0	Average response data sent per second in response to requests
Actual maximum concurrent connections (total 100)	81.0-100.0	96.0-100.0	Actual maximum concurrency under 100 concurrent request conditions
Successfully processed requests	20629	22859	Total number of transactions successfully processed by the system
Number of failed processing	0-60	0-10	Number of requests the system failed to process
Longest transaction time (sec)	35.0-50.0	15.0-40.0	System response request maximum wait time
Minimum transaction time (sec)	0.00	0.00	System response request minimum wait time

## 6. Conclusions

This system is a B2C website. According to its nature, we can apply it to the online shopping mall around Zhengzhou University of Light Industry. After the completion of the system, it can fully assume the use of Zhengzhou University of Light Industry and its surrounding users, and its service expandability is a good advantage. Thus, we can expand the business according to the needs of the corresponding expansion. In addition, the online mall can help the second-hand book market of Zhengzhou Light Industry College solve the rare problem of offline sales. At the same time, in small and medium-sized enterprises, the performance problems of the system will be improved to an extent, and most of the software used in this system is free and open-source. In the later operation process, we only need to consider the server host rental problem. The system uses the Alipay interface of Ant Financial in the payment, which has high security and a wide application range of Alipay, so it is convenient to use. Therefore, the online mall system has a good development prospect.

## Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61672471 and 61572445), Fundamental Research Funds for the Henan Province University (No. 17KYYWF0202), Plan for Scientific Innovation Talent of Henan Province (No. 184200510010), and Henan Province University Science and Technology Innovation Team (No. 18IRTSTHN012).

## References

1. S. Kang, W. Park, and Y. K. Kim, "Dynamical E-Commerce System for Shopping Mall Site Through Mobile Devices," *Lecture Notes in Computer Science*, Vol. 4055, pp. 268-277, 2006
2. H. D. Kim, S. Kim, and Choi, "Web Client System for 3D Virtual Shopping Mall," in *Proceedings of the IASTED International Conference on Internet and Multimedia Systems and Applications*, Vol. 7, pp. 669-674, 2003
3. Z. Nie, "SaaS-based Online Mall Trusted Platform Analysis and Design," in *Proceedings of ICACTE 2010-2010 3rd International Conference on Advanced Computer Theory and Engineering*, Vol. 3, pp. 3639-3643, 2010
4. W. Hong, W. Yang, and L. Xie, "Design and Implementation of Web-based DSS for Online Shopping Mall," in *Proceedings of 2009 IEEE International Conference on Control and Automation*, pp. 1308-1313, 2009
5. D. L. Liang and H. B. Chen, "An Online Mall CRM Model based on Data Mining," *Advances in Intelligent Systems and*

- Computing*, Vol. 510, pp. 599-606, 2017
6. C. H. Tang, J. J. Tang, and C. Liang, "The B2C based on JSP Electronic Mall of Designing and Realizing the System," in *Proceedings of the 2nd International Conference on Electronic and Mechanical Engineering and Information Technology*, pp. 893-896, 2012
  7. T. Ahn, S. Ryu, and I. Han, "The Impact of the Online and Offline Features on the User Acceptance of Internet Shopping Malls," *Electronic Commerce Research and Applications*, Vol. 3, No. 4, pp. 405-420, 2004
  8. Y. J. Chang, F. Y. Leu, and S. C. Chen, "Applying Web Crawlers to Develop a Sale Management System for Online Malls," in *Proceedings of 2015 9th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, pp. 408-413, September 30 2015
  9. B. S. Shim and Y. M. Suh, "CRM Strategies for a Small-Sized Online Shopping Mall based on Association Rules and Sequential Patterns," in *Proceedings of PACIS 2010-14th Pacific Asia Conference on Information Systems*, pp. 355-366, 2010
  10. K. K. Nam, K. Kim, and S. Hong, "A Study on Category Type of Online Shopping Mall," in *Proceedings of the IADIS International Conference Interfaces and Human Computer Interaction 2011*, pp. 503-505, 2011
  11. L. K. Chang and N. Chung, "A Web DSS Approach to Building an Intelligent Internet Shopping Mall by Integrating Virtual Reality and Avatar," *Expert Systems with Applications*, Vol. 28, No. 2, pp. 333-346, February 2005
  12. Y. Hendriana, A. Pranolo, S. Sulaiman, and L. H. Fong, "Generic Shopping Mall Directory Mobile Application," in *Proceedings of 2015 International Conference on Science in Information Technology: Big Data Spectrum for Future Information Economy*, pp. 363-368, February 16 2016
  13. Y. Chu and J. Lee, "The Experiential Preferences of the Online Consumers in Different Internet Shopping Lifestyles Towards Online Shopping Websites," *Lecture Notes in Computer Science*, Vol. 4553, No. 4, pp. 3-11, 2007
  14. Z. Bao and T. Huang, "Exploring Stickiness Intention of B2C Online Shopping Malls: A Perspective from Information Quality," *International Journal of Web Information Systems*, Vol. 14, No. 2, pp. 177-192, June 18, 2018
  15. S. C. Lee, S. W. Kim, and S. Park, "Recommendation in Online Shopping Malls: Results and Experiences," in *Proceedings of the 2013 Research in Adaptive and Convergent Systems*, pp. 116-117, 2013

**Zengyu Cai** received his master's degree in computer application technology from Northeast Normal University in 2006. He is currently an associate professor at Zhengzhou University of Light Industry. His research interests include trusted computing, plan recognition, and information security.

**Yuanbo Liu** is a master's student in the School of Computer and Communication Engineering at Zhengzhou University of Light Industry. Her research interests include content-centric networks and network security.

**Yong Gan** received his Ph.D. in computer application technology from Xi'an Jiaotong University in 2013. He is currently a professor at Zhengzhou Institute of Engineering and Technology. His research interests include information security, cryptography, multimedia communications, and network engineering.

**Jingxiao Li** received her bachelor's degree in computer science and technology from Hebei Finance University in 2018. She is a postgraduate student at Zhengzhou University of Light Industry. Her research interests include content-centric networks and artificial intelligence.

**Yuan Feng** received her master's degree in communication and information systems from Chongqing University of Posts and Telecommunications in 2006. She is an associate professor at Zhengzhou University of Light Industry. Her research interests include mobile communication, network engineering, and information security.