

# The Technical Design and Implementation of Cross-Platform Industrial Product Order System

Yu Xue<sup>1,2,\*</sup>, Xu Cai<sup>1</sup>, Shoubao Su<sup>2</sup>, Junxiang Han<sup>1</sup> and Romany F. Mansour<sup>3</sup>

<sup>1</sup>School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, 210044, China

<sup>2</sup>Jiangsu Key Laboratory of Data Science and Smart Software, Jinling Institute of Technology, Nanjing, 211169, China

<sup>3</sup>Department of Mathematics, Faculty of Science, New Valley University, El-Kharja, 7251, Egypt

\*Corresponding Author: Yu Xue. Email: xueyu@nuist.edu.cn

Received: 02 April 2021; Accepted: 06 April 2021

**Abstract:** According to some data in the Industrial Purchasing Trends report released by China in 2017, we can see that e-commerce purchasing channels have ranked first among all industrial products purchasing channels in China compared with European and American countries. In addition, in the whole industrial product purchasing market, we can also see that both manufacturers and suppliers are making active e-commerce transformation, and some other Internet giants are also actively entering the industrial product e-commerce industry. But at present, the revenue of all kinds of subjects is still a lot of room for improvement compared to the United States industrial giants. Although the domestic e-commerce market of industrial products has a variety of problems, also contains huge opportunities and development space. Today mobile Internet technology is becoming more and more popular. It is particularly important to develop a cross-platform industrial product order system that supports the collaborative work and unified experience of Android, iOS, and Web. This system uses a uni-app framework to develop front-end applications, which can realize an order management system with code running across multiple platforms. The back end is built based on LNMP architecture. Linux is the most popular free operating system. Nginx is a free and efficient web server with good stable performance, rich functions, simple operation and maintenance, fast processing of static files, and minimal system resource consumption. MySQL database is one of the most widely used relational databases in Web application data processing. The server side is written by PHP script under ThinkPHP framework, which is quick, open-source, and cross-platform in system construction. And these four kinds of software are free, open-source software, together, they can become a free, efficient, highly extensible website service system.

**Keywords:** Order management system; cross-platform; LNMP architecture; uni-app framework; ThinkPHP framework

## 1 Introduction

In 2019, the annual purchase amount of MRO industrial products in China has reached 2.11 trillion yuan, with an online penetration rate of about 3.69% [1]. With the improvement of information technology and the popularization of mobile Internet, the online penetration rate of China's industrial product procurement and sales will gradually increase, and the traditional model of industrial product procurement and the mobile Internet industry will be deeply integrated. "Industrial products e-commerce" began to play a more and more role in the field of industrial products sales. There is no doubt that e-commerce sales of industrial products are a new direction for the global industrial layout, which is of great significance to promoting the transformation and upgrading of the manufacturing industry, the



digital economy, and the high-quality development of the economy [2].

The establishment of an e-commerce platform for industrial products is inseparable from technology, platform, network, and infrastructure construction, and involves basic issues such as capital and transaction security [3]. In today's highly popular mobile Internet, in addition to the PC end of the online procurement mode industrial goods procurement also gradually transferred to the mobile end. However, due to the mobile terminal has Android iOS WeChat applet and other terminals, adopting native application development will require multiple sets of code, which will cause very expensive costs [4]. To comply with the trend of industrial products purchasing through the internet, and to solve the disadvantages of small and medium-sized enterprises' lack of website design, effective information display, and order management system, the design and implementation of cross-platform industrial products order management system are particularly urgent [5].

The main research content of this paper is how to implement a cross-platform industrial product order management system running on iOS Android H5 PC browser and WeChat. This system adopts the currently popular uni-app cross-platform development technology to realize multi-terminal operation of a set of codes, and the application developed by it has greatly improved its functions, performance, and user experience compared with the traditional H5 cross-platform technology, which has become the choice of mobile application implementation. The front and back-end interactions are realized through API technology, and the back end USES the lightweight, cross-platform, high-performance features of LNMP framework and ThinkPHP's agile and efficient construction of a Web site for order processing system [6]. This paper mainly includes the following research aspects:

- 1) Order management system function division. The functional and performance requirements of the system are determined by analyzing and studying the requirements of the order system. The target system is mainly divided into the front sales module and the back-order processing module. Each module also contains a lot of sub-functions. I am responsible for the design and implementation of SMS registration and login, order generation (purchase), order cancellation, order inquiry, order delivery, and confirmation of receipt.
- 2) Design of order management system. This article carries on the outline design and the detailed design to the system each module, determines the target system to have which module composition and the relation between these modules, and designs the detailed algorithm and the data structure. This is expressed through architecture diagrams and process design tools.
- 3) Implementation of order management system. Choose the appropriate programming language based on the previous detailed design to design a program that is logical and easy to read and understand. Use the uni-app framework and ThinkPHP framework to implement the front and back-end code. In order to ensure the normal operation of the order management system and to meet all functional and performance requirements, a detailed test case was designed for functional and non-functional testing of the system. Identify and correct potential errors in the code to ensure the quality of the order management system.

## 2 Related Work

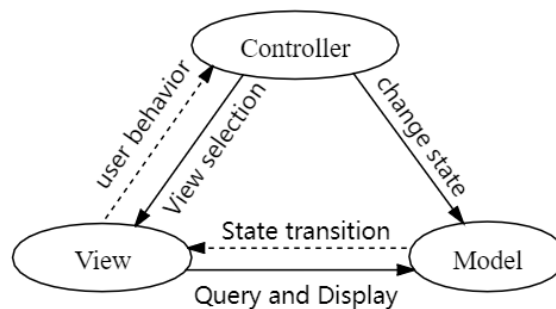
The technical selection of a software project needs to be in line with the development cost and use environment of the software. The technical selection is usually made before the software development and design according to the actual requirements and the technical level of the development personnel. Therefore, the related technology should not only meet the functional and performance requirements of the system but also suit the technical level of the developers. This system is developed for small and medium-sized industrial enterprises, all related technologies have the characteristics of fast, open-source, cross-platform, and so on, which will greatly save the development cost. The front end of the order management system uses a uni-app framework based on Vue.JS to realize cross-platform development cost-saving, and then connects the back end through API technology. The back end is built under LNMP architecture and written in PHP language under ThinkPHP framework [7].

## 2.1 Uni-App Framework

Uni-app is a popular front-end application development framework, a domestic framework produced by DCloud (Digital Heaven (Beijing) Network Technology Co., Ltd.). It can write a set of code, and then run in multiple terminals, such as IOS Android H5 WeChat Alipay Baidu small programs Toutiao, etc., to solve the problem of multi-terminal repetitive development; Besides, the low-cost learning developer of uni-app can realize cross-platform front-end programming by simply using vue syntax + WeChat applet tags and APIs. So, this is very friendly to beginners or newcomers who have a little bit of a front-end foundation. Of course, it can also realize the call of the platform's proprietary functions through conditional compilation + API calls specific to each platform while realizing the cross-terminal, without affecting the operation of other platforms. Also, uni-app supports native code-mixing for all platforms and native SDK integration for faster page loading and refreshing and is made up of thousands of plug-ins. Greatly reduced development costs.

## 2.2 ThinkPHP Framework

ThinkPHP framework is a domestic lightweight development framework, it has rapid development, strong compatibility, and the characteristics of simple and easy-to-use [8], its purpose is to simplify the development, improve efficiency, easy to expand, and support a variety of database of popular now, because ThinkPHP framework it is domestic has a wealth of documentation and examples for reference for developers. ThinkPHP shown in Fig. 1, is MVC (short for Model-View-Controller) mode, which is easy to understand and easy to get started with [9]. The framework design of MVC can promote the sharing between the code and enhance the program's extensibility. This pattern divides the system into three modules: model, controller, and view, and separates the output, processing, and output. ThinkPHP's most attractive feature is the ease and speed with which you can build a website.

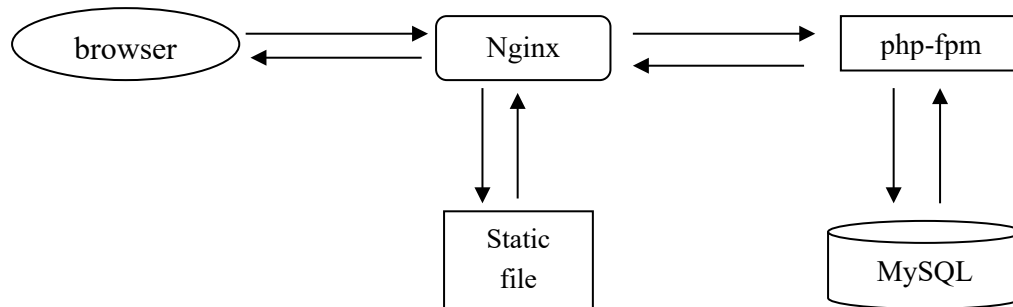


**Figure 1:** MVC realization schematics

## 2.3 LNMP Framework

LNMP refers to the website server architecture that builds three free software Nginx+ MySQL+ PHP under the Linux system. Nginx is a high-performance HTTP server and can implement reverse proxy, of course, can also do IMAP/POP3/SMTP proxy server. MySQL is a free, widely used small relational database management system. PHP is a scripting language that executes on the server-side and can be embedded in HTML documents. Nginx server is relatively stable, with rich functions, simple installation and configuration, and a low system resource occupancy rate. Nginx can support PHP directly internally, and can also support external service as HTTP proxy server [10]. Nginx is written in C and is better at both system resource overhead and CPU usage than Apache. Using this architecture will improve the performance of the site in many ways. As shown in Fig. 2, the LNMP combination is favored by many Internet full-stack developers due to its advantages of open source, reliability, and simplicity. The integration of these four technologies is the result of the free combination of the open Internet environment.

The main difference between LNMP and LAMP is N (Nginx). Nginx is a free and efficient Web server and has become the second-largest Web server in the world after Apache [11]. With its support for high concurrent access and proxy capabilities, Nginx can quickly parse static files. It USES asynchronous sockets to receive client requests, which reduces CPU and system resource overhead and is much more efficient than Apache. In addition, Nginx has the reverse proxy capability to resist external attacks on the server, thus maintaining the security of the Internal Web. Fig. 1 shows the overall performance comparison between the Nginx server and the Apache server.



**Figure 2:** Programming environment setup of LNMP

**Table 1:** The performance comparison of Nginx and Apache

Webserver	Nginx	Apache
reverse proxy	very good	good
Rewrite regulation	very good	good
FastCGI	good	bad
Hot Deploy	support	nonsupport
System pressure	very small	small
stability	very good	good
security	general	good
technical literature	less	much
Assetic	very good	general
Virtual Host	support	support
memory consumption	very small	very big

### 3 Design

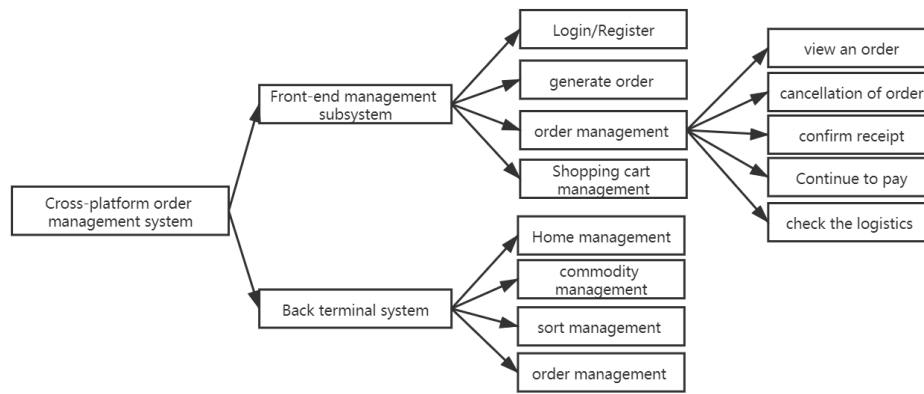
#### 3.1 Design of System Architecture

LNMP architecture (Linux+ Nginx+ MySQL+ PHP) and another LAMP architecture (Linux+ Apache+ MySQL+ PHP) will be adopted for the back-end servers of this system. The biggest difference between them is the difference between Web servers. Although Apache is a mature and stable Web server, and it is widely used, it occupies a large number of resources, runs slower than Nginx, and has poor load and concurrency. Nginx occupies a small number of resources, has better concurrency processing capability, supports reverse proxy, and has good load performance.

This system chooses a uni-app framework to develop a cross-platform industrial product order management system. Uni-app is an open-source framework that can be applied to vue.js. Uni-app is friendly to front-end developers with low learning costs. Therefore, the front end of the system will use a uni-app framework to develop and use its request to access API interface to obtain back-end data to realize the separation of front end and back end and utilize Uni-app to realize the cross-platform feature of the system [8].

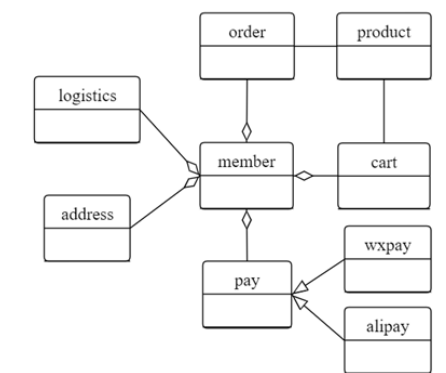
### 3.2 Design System Functional Structure

The overall functional structure of the system is designed from the perspective of the system’s goals and the actual performance requirements. The main purpose of system function design is to reasonably divide the system into several function modules based on-demand analysis and to determine the call relationship between each module and their data relationship [12]. The functional structure diagram of the cross-platform order management system is shown in Fig. 3.



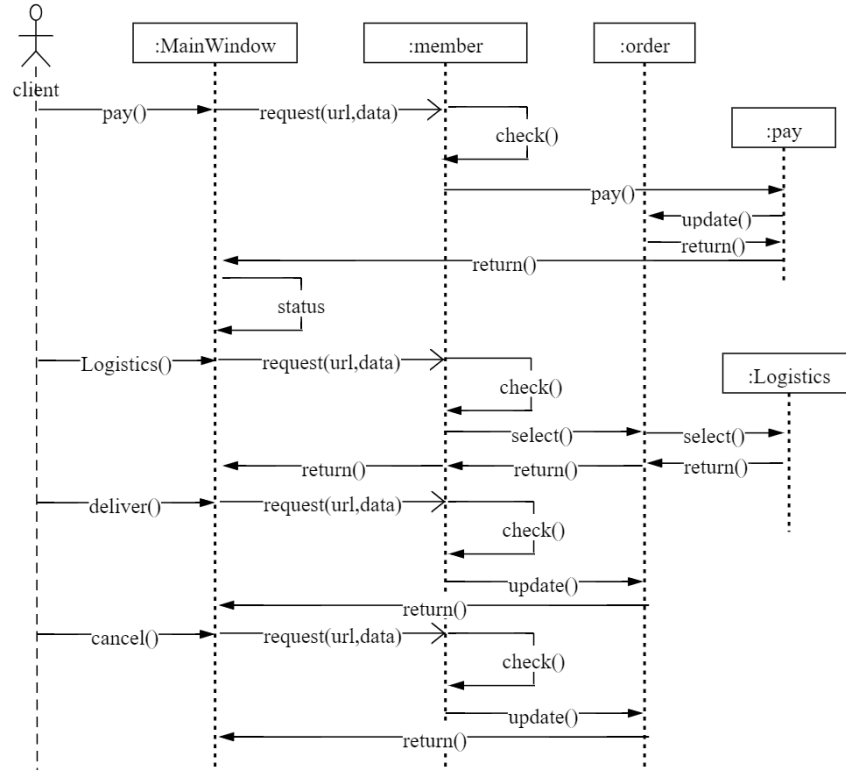
**Figure 3:** The functional structure of the system

The main class diagram design of the system is shown in Fig. 4. The Member class can be called when the front-end requests after it has operated on the shopping cart. Cart class to realize the corresponding operation of the shopping cart in the database; You can also call the order class to generate the order when the order is generated. There are many other methods in the order class, including confirming receipt, canceling, and so on. The pay class is called when the order is paid, and it has two subclasses, wxpay (WeChat payment) and alipay (Alipay payment). In consideration of decoupling and future maintenance, we will extract the payment Pay method instead of putting it in the order class, which will facilitate maintenance if new payment methods are added in the future. The logistics class is used by the Logistics class to operate the Logistics information in the database, and the Address class encapsulates the operation of the customer's receiving address.



**Figure 4:** The system class diagram

Order management is the core of the management system. The normal operation process is shown in Fig. 5 Customers can continue to pay for the order, check the logistics, confirm receipt, cancel the order, and other operations after viewing the order. Each time the corresponding operation is selected, the Token will be verified to confirm the identity, and the verification pass will call the corresponding method to change the information of the database, and then return the information of the corresponding operation.



**Figure 5:** Order management module

#### 4 Implementation and Testing

According to the design and function of the system, this chapter will implement and test the main functions of the front and back end of the order management system. This includes login registration, homepage, membership interface, shopping cart, order generation, payment, order viewing, address management, and so on. The implementation part reflects cross-platform performance. Front-end testing used postman's tool for functional testing of the interface.

The front-end login page and the registration page are shown in Fig. 6. New users can sign up for an account via SMS and then log in to the system for personal operations.

The home page and shopping cart are shown in Fig. 7. Open the program, and you can display it. However, the shopping cart cannot be used until you are logged in, and will return different information depending on the authentication.

The interaction between front and back ends is very important in the system, and the information interaction between front and back ends is realized through URL. The header information for the login request is shown in Tab. 2.

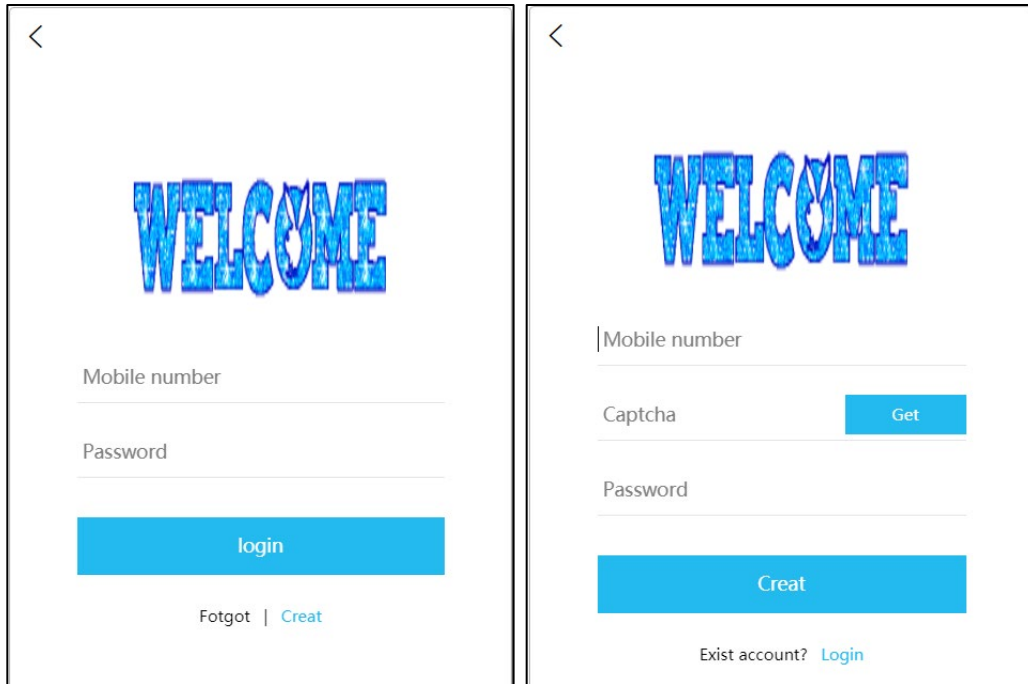


Figure 6: Login and registration page

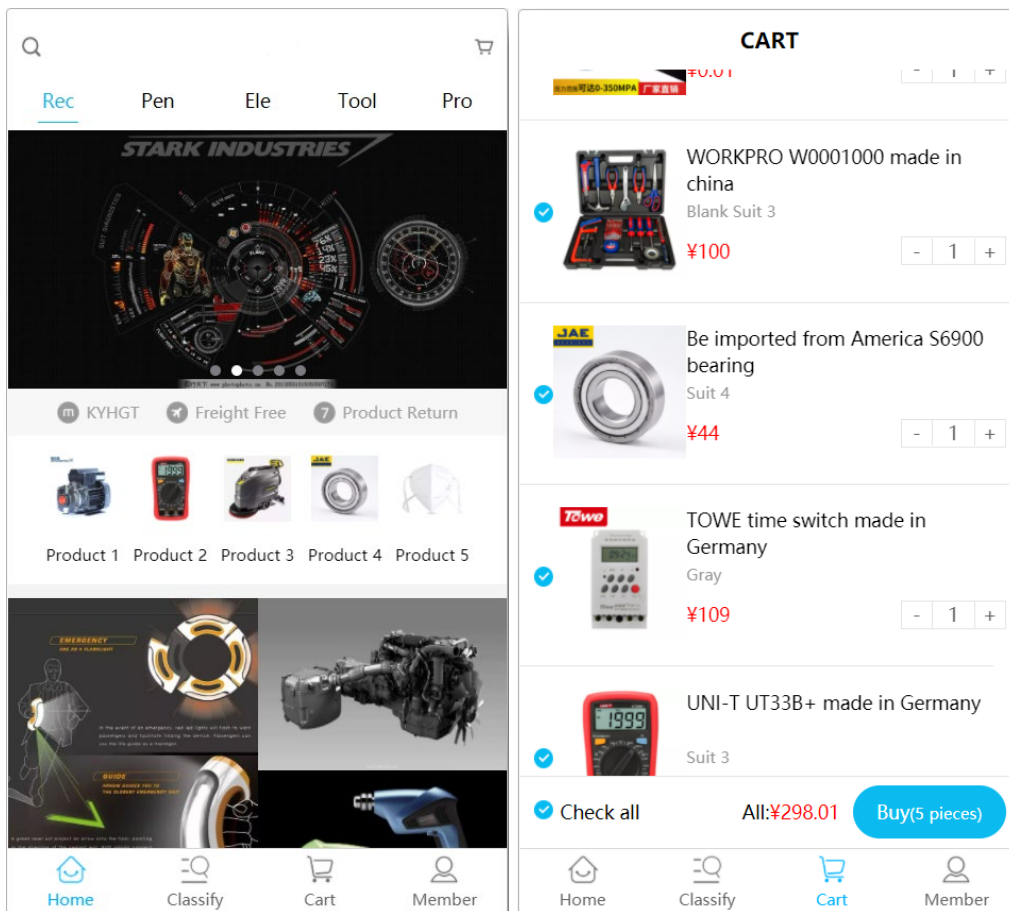


Figure 7: Home and shopping cart page

**Table 2:** The header information for the login request

ParamName	interval	type	explain	error code	explain
telephone	yes	string	Phone number	0	failure
password	yes	string	password	1	succeed

An example of the information returned

```
{ "code": 1,
  "data": {
    "Token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.eyJ1aWQiOiJlYmVudC50dG1OTA0Nn0.MZ8Dnolbek8dFQoVOTeM7ZwgdN4gxsu_YeglZSfYFSg"
  },
  "msg": " succeed "
}
```

The backend mainly tests the response speed of the system. Hardware, software, and network will all affect the speed of front-end requests. JMeter was used to simulate the concurrent load environment of the system and detect the carrying capacity of the system, and the system deployed on the server under the 4 M broadband network. The configuration of the server is 2 core CPU, 4 G memory, CentOS8.0, and 1 Mbps bandwidth.

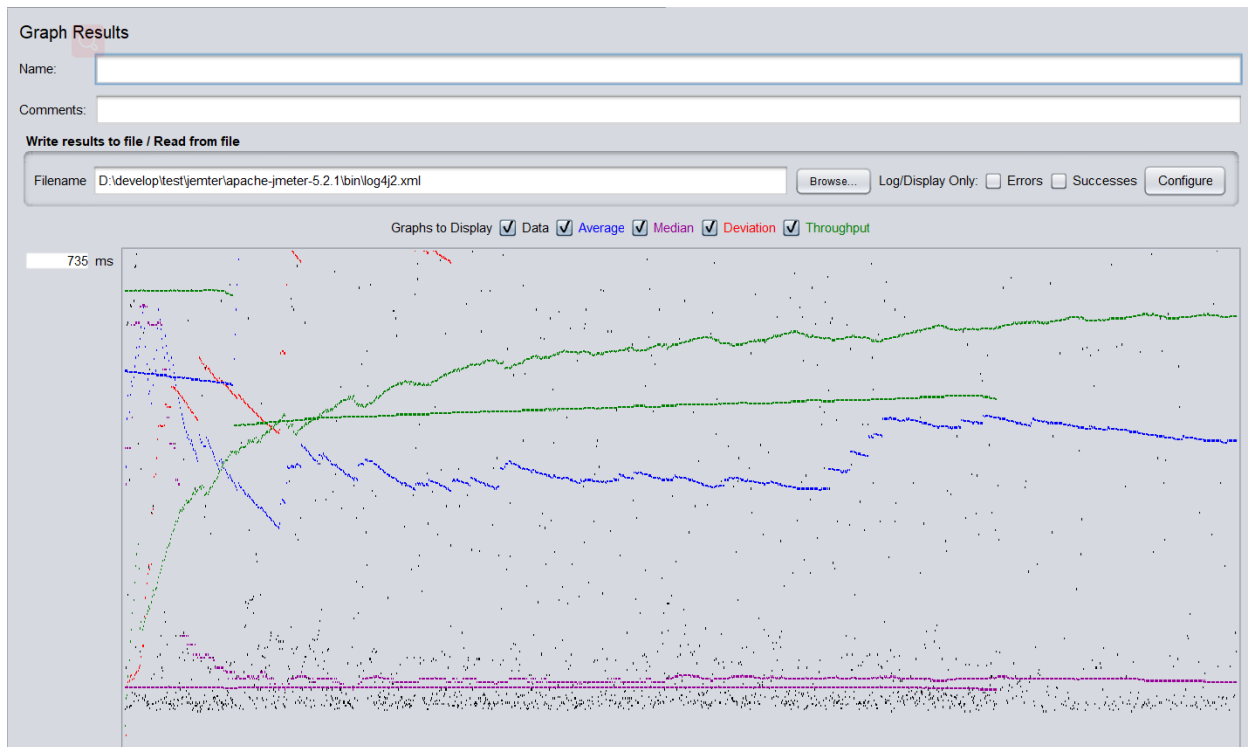
Fig. 8 shows an aggregated report of the interface stress tests, with each interface simulating 20 users with 20 iterations per user for a total of 300 requests. From the diagram we can see the median average response time of each interface, the user response time (50%), 90%~99% percentile (90%~99% of the user response time), the minimum response time, maximum response time, abnormal % error probability (tests) and throughput (by default to complete the number of requests per second), receiving KB/S and send KB/S.

Label	# Samples	Average	Median	90% Line	95% Line	99% Line	Min	Maximum	Error %	Throughput	Received KB...	Sent KB/sec
Home	400	3359	580	2422	8639	77640	82	78115	0.00%	4.2/sec	48.52	1.06
Classify	400	1131	325	1314	2230	34743	75	35579	0.00%	4.2/sec	26.27	1.06
Detail	400	308	139	618	852	2113	73	8662	0.00%	4.2/sec	13.62	1.10
Search	400	106	83	139	330	407	64	1017	0.00%	4.2/sec	3.87	1.08
Cart	400	83	73	103	132	321	61	366	0.00%	4.2/sec	2.44	1.44
Login	400	93	82	115	140	345	66	597	0.00%	4.2/sec	2.43	1.58
Order	400	86	74	116	139	332	61	352	0.00%	4.3/sec	2.44	1.44
TOTAL	2800	738	94	676	1226	8639	61	78115	0.00%	29.1/sec	98.57	8.60

**Figure 8:** JMeter stress test aggregation report display diagram

Fig. 9 is the stress test graph results chart, it mainly presents the throughput, average, median, and deviation in a more intuitive way, so that users can see the trend of the variation of each value.





**Figure 9:** JMeter pressure test graph results display diagram

## 5 Conclusion

The system mainly includes front-end login registration (SMS registration), order generation, order payment, and order view (including confirmation of receipt, payment, logistics view), as well as back-end homepage management, classification management, commodity management, and order management. Under the guidance of my teacher, I completed the development of the system. The backend of the system is deployed on the server of aliyun and adopts LNMP architecture. It USES its lightweight, cross-platform, high-performance, and other characteristics to build an open-source free architecture. In addition, the back end is written with ThinkPHP framework PHP language, while the front end is written with a uni-app cross-platform framework to invoke back-end database resources through API. The cross-platform of the system is mainly realized by the features of the uni-app. At present, it is realized across the android app, Web page, H5, and WeChat small program platform. In which, all functions can be realized on the android app, and some functions of other platforms have not been realized.

Small industrial manufacturers of this system are for the sake of development, has now completed the online deployment and test, but the company in the development of business of the system is also on the rise, members will be more and more, the system still has a lot to improve, for example, third-party login function, more payment, customer service function, relevant features, and so on. The system has just been developed and there is still a lot of room for improvement. In the future operation and maintenance process, the system will be gradually improved and run steadily.

**Funding Statement:** This work was partially supported by the National Natural Science Foundation of China (61876089, 61876185, 61902281, 61375121), the Opening Project of Jiangsu Key Laboratory of Data Science and Smart Software (No. 2019DS301), the Engineering Research Center of Digital Forensics, Ministry of Education, the Science and Technology Program of Jiangsu Province Construction System (2020JH08), and the Priority Academic Program Development of Jiangsu Higher Education Institutions.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] H. D. Lu, "Marketing network of marine products under e-commerce mode," *Coastal Education & Research Foundation*, pp. 267–271.
- [2] A. Erdmann and M. P. José, "Digital inbound marketing: Measuring the economic performance of grocery e-commerce in Europe and the USA," *Technological Forecasting and Social Change*, vol. 162, no. 120373, pp. 1–13.
- [3] C. Huang, J. Zhao and Y. Cui, "Research on recommendation systems in the electronic commerce," *2018 Chinese Control and Decision Conference*, Shenyang, pp. 1476–1481.
- [4] Y. Zhang, "The implementation of an English word learning system feedback system and smartphone app," *Computer Systems Science and Engineering*, vol. 35, no. 3, pp. 207–214, 2020.
- [5] C. F. Wang, X. J. Fan and Z. Yin, "Financing online retailers: Bank vs. electronic business platform, equilibrium, and coordinating strategy," *European Journal of Operational Research*, 2019.
- [6] Z. W. Tian, S. B. Tian, T. Wang, Z. Gong and Z. Q. Jiang, "Design and implementation of open source online evaluation system based on cloud platform," *Journal on Big Data*, vol. 2, no. 3, pp. 117–123, 2020.
- [7] W. Hui, G. Y. Sun, Q. Y. Zhang, K. M. Liu, M. Xi *et al.*, "The design and implementation of a service composition system based on a restful API," *Intelligent Automation & Soft Computing*, vol. 25, no. 3, pp. 573–583, 2019.
- [8] K. A. Gary, S. Sohoni and R. Acuna, "Scaling to meet the online demand in software engineering," *International Journal on Innovations in Online Education*, vol. 4, no. 1, 2020.
- [9] H. Wang and F. Fang, "Research on e-commerce supply chain design based on MVC model and virtual image technology," *IEEE Access*, vol. 8, pp. 98295–98304, 2020.
- [10] D. L. Paris, M. Bahari and N. A. Iahad, "Business-to-customer (B2C) Electronic Commerce: An implementation process view," in *2016 3rd Int. Conf. on Computer and Information Sciences (ICCOINS)*, pp. 19–24.
- [11] F. Igor, B. Radovan and G. Beata, "Technology acceptance model in e-commerce segment," *Management & Marketing*, vol. 13, pp. 1242–1256, 2018.
- [12] K. H. Leung, K. L. Choy, P. K. Y. Siu, G. T. S. Ho and C. K. M. Lee, "A b2c e-commerce intelligent system for re-engineering the e-order fulfilment process," *Expert Systems with Applications*, vol. 91, pp. 386–401.