Design and Implementation of Online Education SaaS System Based on Microservice Architecture

Zhuhui Tao[1](✉) and Hong Wang[2]

[1] Xi'an Shiyou University, No.18, East Section of Electronic Second Road, Xi'an, Shaanxi, China
taozhuhuii@163.com

[2] School of Computer Science, Xi'an Shiyou University, No.18, East Section of Electronic Second Road, Xi'an, Shaanxi, China

Abstract. In the process of digital transformation of education, online education system plays a key role in promoting the digital development of education. In response to the current problem that the traditional monolithic architecture is too tight and cannot meet the usage requirements in terms of performance, security, availability, scalability and other attributes, this paper discards the traditional monolithic architecture and proposes an online education SaaS system developed based on Spring Cloud Alibaba microservice architecture; firstly, based on the analysis of monolithic architecture and microservice architecture, this paper selects the micro Secondly, after analyzing the traditional application system and SaaS system, the SaaS system is selected; next, the architecture of the online education SaaS system is designed and built; finally, the functions of the online education SaaS system are completed based on the architecture. Finally, a complete and feasible online education SaaS system based on microservice architecture is realized.

Keywords: Spring Cloud · Microservices · SaaS Platform · Online Education

1 Introduction

With the development of education informatization, education mode began to "Internet + education" change, the advantages of online education convenient and fast more obvious. In recent years, online education has continued to develop, and more and more enterprises and institutions have started to transform their education. Online education systems have gradually entered a mature stage from the beginning of project-based software, packaged software to platform-based software, but the disadvantages also come with it. China has millions of small and medium-sized enterprises, along with higher human and material costs, most small and medium-sized enterprises can not afford the cost of development and operation and maintenance, and the traditional monolithic architecture [1] has been unable to meet the high concurrency problem. Therefore, how to solve the problem of concurrency avalanche caused by too many users and develop an efficient and wide range

D. Kumar et al. (Eds.): IEIT 2023, AHSSEH 10, pp. 481–488, 2023.
https://doi.org/10.2991/978-94-6463-230-9_57

of users of online education system under the premise of high efficiency, stability and security [2] is an inevitable trend in response to the development of national education informatization.

The current research of online education system, Renping Xie and others [3] proposed a front and back-end separation development model based on Vue and SpringBoot, and developed an online education system using traditional MVC architecture, but the application scope is small. Baolong Wang [4] designed and implemented an online education system based on cloud computing, chose SaaS model, combined with traditional architecture, and the main function realized online answering. Jiapei Wang [5] proposed an online education system based on microservice architecture, integrating SpringMVC and microservice technology to achieve cluster deployment, with better system functions, but unable to cope with more complex usage scenarios of enterprises.

To meet the usage needs of enterprises, a SaaS system based on microservice architecture is a good solution, i.e., one platform serves multiple customers, which enables software enterprises to expand the usage scope of users and enhance the flexibility of business transformation. In this paper, we transform from a monolithic structure that can only serve one user, design and implement an online education SaaS system based on microservices architecture, utilize the characteristics of independent database data space, realize an effective data transaction processing mechanism, ensure the security of data, and solve the user volume and performance problems.

1.1 Microservice Architecture Features

For the traditional monolithic architecture, the interdependence between applications is too high, and it is difficult to scale. When deploying, all modules need to be uploaded to tomcat as war packages, and it is difficult to achieve load balancing, which requires more wasted resource allocation [6]. On the one hand, it is caused by the growing complexity of the system, and on the other hand, it is caused by the characteristics of the system that affects the whole body. Microservice architecture is different from the traditional monolithic software architecture, and is a software architecture created to meet the “three high demands” (high concurrency, high performance, and high availability) of the current Internet backend, which can be said to be an architectural model that decomposes functions into a series of services, and each service communicates with each other through a “The main features include:

1) Isolation. A single application is divided into a variety of small, interconnected microservices, a microservice to complete a relatively single function, independent and decoupled from each other, the unavailability of one service will not cause another service is also down.
2) Technical heterogeneity [7]. The development technologies within different services can be inconsistent, and the most suitable technologies and development languages can be selected for different services, and different storage technologies can be used for different parts of the system, making the development more flexible and the system more fault-tolerant.
3) Scalability. You can do expansion and upgrade of services that affect performance, and play the effect of the right medicine.

1.2 Features of SaaS

SaaS is a software delivery model that delivers software to users as a service. Users no longer buy software, but rent web-based software and pay for it according to their use of the software, realizing a variety of uses through a multi-tenant model to meet the individual needs of different users [8]. SaaS is characterized by the following:

1) Cloud deployment. SaaS is an application-based product based on cloud services, which provides services to users in the form of Internet browser or Web Services connection.
2) SaaS is a standardized product, not a customized software.
3) Multi-tenant architecture. SaaS is usually based on a standard software system that provides services to hundreds of different customers, supporting the isolation of data and configuration between different tenants.
4) Subscription-based payment. Unlike the traditional software one-time license payment, SaaS subscription is the core of the "on-demand, pay-as-you-go".
5) SaaS provides services to enterprises and organizations of all sizes, often as customizable, large-scale business solutions.

2 Proposal Method

2.1 Architecture Design

For the integration of microservice architecture, the mainstream are Spring Cloud, Spring Cloud Alibaba and Dubbo, in comparison, Dubbo is a distributed RPC framework, which only addresses the service discovery, service routing, service degradation, load balancing capabilities of the registry, and is not a complete ecology, while Spring Cloud and Spring Cloud Alibaba community is more active, the ecological mechanism is more complete, convenient integration of the registry, configuration center, gateway, load balancer, fusion mechanism, service data monitoring and other components [9], including Spring Cloud Consul container can be seamlessly integrated with Docker, more rapid and effective development of software management. It can be seen that Spring Cloud Alibaba ecosystem is well configured for development and operation and maintenance, integrates and inherits Spring Cloud solutions and provides a better extension to the microservices framework, the more obvious of which is nacos, which enables online configuration, saves more development resources and handles zero code servers. Therefore, this paper selects Spring Cloud Alibaba architecture to implement an online education SaaS system.

In order to better maintain the project and improve the development efficiency, the system uses the technology of front and back-end separation [10], the purpose of separation is to better realize the front and back-end decoupling, the overall software architecture diagram is shown in Fig. 1:

Front-end choice of Vue3.0, Vite, Ant-Design-Vue3, TypeScript technology solutions, Vue as a progressive framework for building user interfaces, is an improved version of Model-view-Controller, that is, the model-view-view model, when the view changes, Vue will achieve two-way binding of data, through the Vue.js for component-based development, saving more code, corresponding to the division into two ends, the back-end and the user side, the two ends have their own duties do not affect each other. In

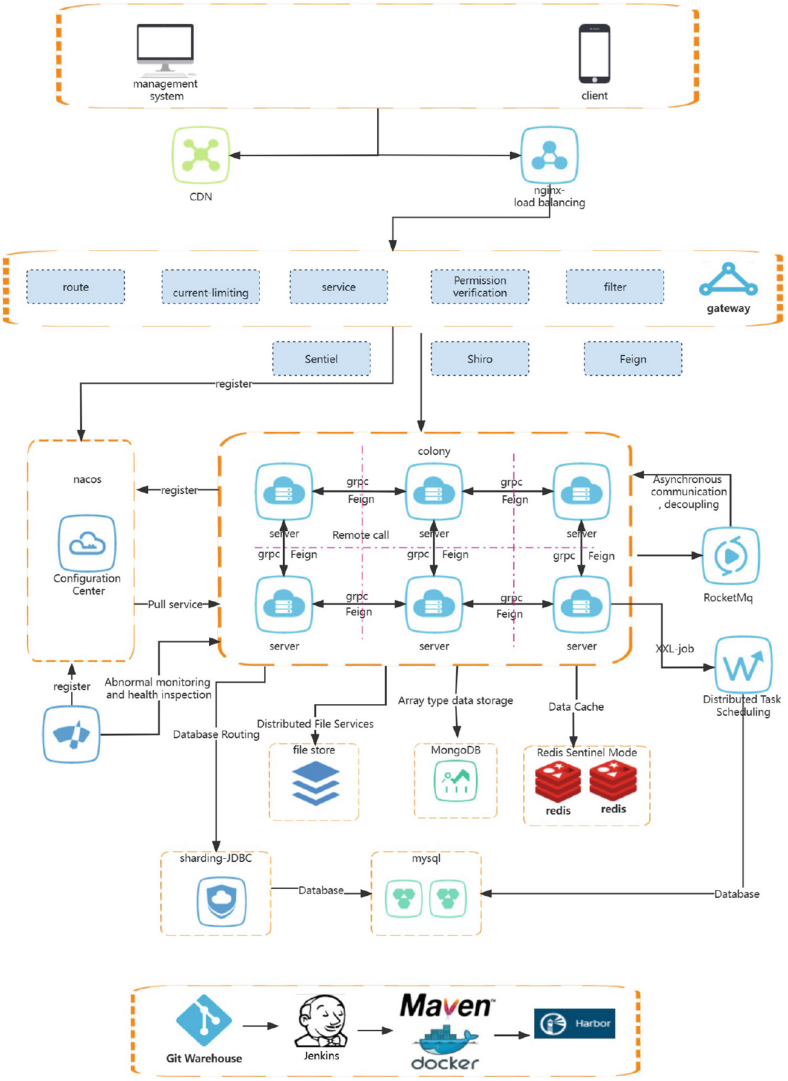

Fig. 1. Overall Architecture Diagram

order to reduce the pressure of accessing the server when the front and back ends of the original monolithic architecture are not separated, the front-end server uses nginx, a high-performance HTTP and reverse proxy web server. Spring Cloud Alibaba microservices architecture used on the back end, multi-module separation, the overall architecture is divided into: gateway module, backend management module, client module, payment module, message notification module, public module, timed task module, open interface module, log service module; The detailed technical descriptions used are as follows:

1) Gateway module: This module is also known as the unified route, the manager of all modules, interface access will go through the interface in order to access, mainly to assume the role of route forwarding, interface authentication, there is no need to configure each module after the gateway, unified management of the API, able to match any request attributes.
2) Call between modules: use Spring Cloud Alibaba ecology comes with Feign, as a bridge between modules to communicate, solve the same registration center, call the interface of different modules, just like this project call the same performance, such as: backend services call the interface of the public module.
3) Business logic layer: different business logic is developed in modules, classifying interfaces for easy maintenance, integrating mybatis multiple data sources, using different databases for different tenants, and achieving data separation before tenants.
4) RocketMq service notification: the client implements a certain scheduling requirement and sends it directly through the producer consuming it without the need for calls between interfaces, which greatly improves the decoupling of code as asynchronous execution so that this does not affect the execution of the next program code.
5) Service registration and discovery: nacos carries both the registration and configuration center of the project, which exists as a very important component in the ecology and is an indispensable part of the project, allowing visual online management of the code configuration, without the need to modify the configuration file inside the code each time, and the association can be formed between different modules registered to nacos, which can be managed in real time after viewing in the page Registration service list module.
6) Data level: relational database using Mysql + Sharding-JDBC to achieve multi-tenant sub-base and sub-table, non-relational database using Redis, MongoDB, redis access is based on memory type, speed performance faster than relational database, MongoDB storage data format supports array format.
7) Monitoring and fault tolerance: Sentinel realizes real-time inspection and control of services, such as the link of service layer invocation, the invocation relationship of resources between service layer and service layer, realizes the health monitoring of system load of the system, and takes the initiative to downgrade if the service is unstable to avoid blocking.
8) O&M Deployment: Building CI/CD Streamlines.

Gitlab + Docker + Jenkins + Harbor to achieve automated deployment, rapid integration of development, saving time for each package deployment [11], nginx to achieve load balancing of the project to prevent redundancy of the project, traffic distribution to multiple cloud servers to maximize the system's high availability.

2.2 SaaS System Design

The system as a whole is divided into three major parts, namely: service providers, tenants and trainees. The service provider mainly manages the tenant information and is the vendor of SaaS in the system; the tenant is the leaser of the product, who can enjoy the corresponding services of the platform after purchasing, and the system of each tenant

is independent of each other; the student is the user who learns on the corresponding service after the tenant rents the corresponding service.

The system's database is based on sharding-JDBC to achieve dynamic switching of data sources, regardless of how many data sources are created for tenants to use, are unified to sharding-JDBC to manage, so as to achieve multi-tenant storage, the dataSourceMap encapsulation of a new dataSource to Spring to manage, different tenants use different databases, and each tenant can distinguish which database to use by the domain it accesses.

3 Result

According to the design of the microservice architecture and database, the final implementation of the online education SaaS system is shown in the following Fig. 2:

1) Agency management: Agency management that is, tenant management, management of each tenant independent domain name and independent database, tenants rented by the system internal deployment of resources, the use of resources between tenants, between users do not affect each other.
2) System management: management roles between the allocation of resources, such as menu management, departmental management, data management, the allocation and management of the system's existing resources, between the roles through permissions control, to ensure the legitimacy of user operations.
3) Module plug-in: as a module for the system to allocate resources for tenants, it is an indispensable main function of the tenant module, and the system needs to allocate the system privileges for tenants to use in order to meet their individual needs.
4) Resource Center: As a module supplemented by market resources, it provides resources for courses, exams and other modules for purchase. Tenants can quickly integrate the content into their own tenant system through the Resource Center to realize the system online in the fastest way and also enrich the overall architecture of the system.

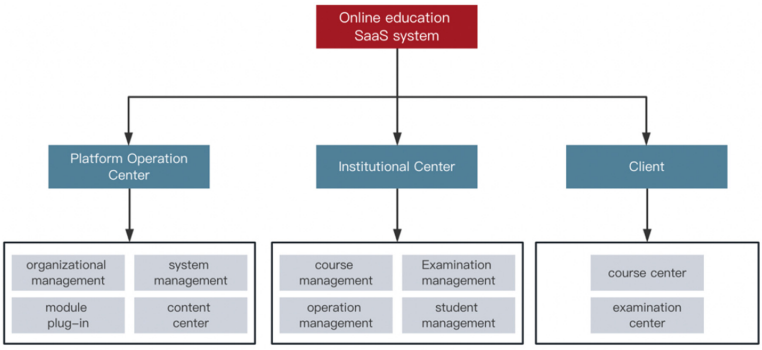

Fig. 2. System overall function diagram

5) Course management: This is the main module of the online education system, on which tenants can publish their own course content for students to learn, including course names, course chapters, course videos, etc. Prices can be set for courses, and users can purchase courses to realize the system's revenue.
6) Exam management: Manage the exams created by tenants for students, the exam categories are divided into mock exams and formal exams, and the administrator can customize the settings according to the course scenarios.
7) Operation management: Tenants decorate their own platform, such as rotating images and theme colors of pages, to enhance the beauty of the system and thus improve the experience of students.
8) Learner Management: Manage and monitor the existing learners in the system, you can manage the information and behavior of the learners, the behavior includes such as class records, study records, exam records, etc.
9) Course Center: Students can learn through the videos released in the course, and get the learning rights of the course through the rules set by the administrator in the background management, such as whether they need to pay for it, and students can view their own learning records and book courses [12].
10) Exam Center: Students can participate in exams to make a summary of a certain stage of learning, and after the completion of the exams, students can view their exam records and wrong question records to check and fill in the gaps.

4 Conclusion

This paper presents an online education SaaS system designed and implemented based on Spring Cloud Alibaba microservice architecture, which rationalizes and splits the business modules of the system, uses microservice architecture technology combined with SaaS model, incorporates load balancing mechanism, gateway controller, etc. to achieve a complete component-based ecosystem, ensures data security while reducing operating costs, achieves a great improvement in system performance, and enhances system intelligence and stability.

References

1. Hong, P.H. Mao, W.Q. Qu, S.C. (2021) Design and implementation of an online education platform based on microservice architecture[J]. Computer Knowledge and Technology,17(04):10-12.DOI:https://doi.org/10.14004/j.cnki.ckt.2021.0305.
2. Ren, L.S. (2022) Web-based online education system design and implementation[J]. Computer Programming Skills and Maintenance,(11):24–26+46. DOI:https://doi.org/10.16184/j.cnki.comprg.2022.11.014.
3. Tao, M. Xie, R.P. (2022) SpringBoot-based online education system development and application practice[J]. Software Guide,21(07):170–174. https://kns.cnki.net/kcms2/article/abstract?v=3uoqIhG8C44YLTlOAiTRKibYlV5Vjs7iJTKGjg9uTdeTsOI_ra5_XZf9igzsls-kLD8TiCocopA5RkGI01ASGZuQVr1sHL-x&uniplatform=NZKPT&src=copy
4. Wang, B.L. (2022) Research and implementation of cloud-based online education platform system[J]. Electronic Technology and Software Engineering,(15):220-223.

5. Wang, J.P. (2020) Design and implementation of online education system based on microservice architecture[D]. Huazhong University of Science and Technology. DOI:https://doi.org/10.27157/d.cnki.ghzku.2020.004668.
6. Mao, Y.Z. (2020) Design and implementation of a microservices-based online education system [D]. Huazhong University of Science and Technology, 2020. doi:https://doi.org/10.27157/d.cnki.ghzku.2020.002445.
7. Zeng, J. (2023) Design of microservice architecture based on Spring Cloud [J]. Electronic Technology, 52(01):54–55. https://kns.cnki.net/kcms2/article/abstract?v=3uoqIhG8C44YLTlOAiTRKibYlV5Vjs7ioT0BO4yQ4m_mOgeS2ml3UEFhglMbKAOrD8me_l8106nIfywUzDd8R49g1m7nf0ET&uniplatform=NZKPT&src=copy
8. Rao, C.L. (2021) Research and implementation of a customizable SaaS platform based on microservice architecture[D]. Xi'an University of Electronic Science and Technology. DOI:https://doi.org/10.27389/d.cnki.gxadu.2021.002247.
9. Wang, C. Li, J.J. Zhu, J.J. (2021) Design and implementation of SpringCloud-based online video education system [J]. Digital Technology and Applications,39(12):198-200.DOI:https://doi.org/10.19695/j.cnki.cn12-1369.2021.12.64.
10. Xue, Y.L. Huang, J.H. Shao, T.J. (2022) Research and design of online course learning system with microservice architecture[J]. Computer Age,(05):130–133+137.DOI:https://doi.org/10.16644/j.cnki.cn33-1094/tp.2022.05.034.
11. Cui, H. (2021) Research and application of online education system based on distributed cloud computing [D]. East China Normal University. DOI:https://doi.org/10.27149/d.cnki.ghdsu.2021.000782.
12. Cui, T.M. Liu, W. (2021) Design of personalized online education system based on big data technology[J]. Modern Electronics Technology, 44(05): 175-180. DOI:https://doi.org/10.16652/j.issn.1004-373x.2021.05.036.