Research on the Application of DevOps in the Smart Campus of Colleges and Universities

Lin Pingrong¹, Shi Xiaoquan¹, Yang Junqin¹

¹ Institute of Software, Software Engineering Institute of Guangzhou, Guangzhou, Guangdong, 510900, China

*Corresponding author’s e-mail: 13480387056@163.com

Abstract: With the advancement of the construction of smart campuses, legacy systems coexist with new systems, and frequent updates and iterations of new and old systems, which consume a lot of manpower and material resources. In order to meet the needs of a fast iterative system, a set of DevOps application schemes based on smart campuses are proposed. The plan fully considers legacy system transformation scenarios, and builds the DevOps platform through ZenTao, Jenkins, SonarQube, Elastic, Docker, Zabbix and other technologies to provide a good communication channel for development and operation and maintenance teams, reduce development iteration costs, and realize automated management. Practice has proved that the solution greatly shortens the delivery time of the system and improves the delivery quality, which can continuously promote the development of products and facilitate the construction of smart campuses.

1. Introduction

With the implementation of the "Education Informatization 2.0 Action Plan", the construction of domestic education information has gradually entered a new stage. New information technologies such as artificial intelligence, big data, and the Internet of Things will lead the transformation of various business scenarios on campus to intelligence. The 13th Five-Year on National Informatization Plan clearly pointed out that colleges and universities should grasp the trend of technological change, consolidate their efforts to build smart campuses, and promote the intelligent transformation of education [1]. With the advancement of the construction of smart campuses, the number and scale of systems for teaching and management in colleges and universities are increasing, and the number of teachers and students in colleges and universities is also increasing. With the changes in education policies, demands and changes from transition to digital campus to smart campus[2] are becoming more frequent, and the generation of new systems or the iteration of old systems is inevitable. Whether it is the development of a new system or the frequent update and iteration of the old system, it will bring huge costs on demand and change and maintenance. Therefore, it is necessary to construct a fast, convenient, safe, and iterative development plan that can save manpower and material resources. Based on this scenario, this article focuses on the application research of DevOps in university smart campuses, and builds a specific and feasible DevOps platform to reduce the cost of development iterations and improve efficiency. High-efficiency iteration allows products to be launched and released earlier, and at the same time, it is easy to make trial and error quickly, and iterate products quickly through user feedback data, which is conducive to promoting the construction of smart
2. Introduction to DevOps

DevOps is a combination of Development and Operations. It is a collective term with a set of processes, methods and systems. It is used to promote communication, collaboration and integration between development, technical operations and quality assurance departments. It is a software development method that involves software in the part of continuous development, continuous testing, continuous integration, continuous deployment, and continuous monitoring in the development life cycle, which helps more communication and collaboration between the development team and the operation and maintenance team for the purpose of creating better quality software, quickly and reliably [3-4]. By automating the process of "software delivery" and "architecture change", software can be built, tested, and released more quickly, frequently, and reliably. Its emergence is due to the increasingly clear realization of the software industry: in order to deliver software products and services on time, development and operations must work closely together.

Compared with the traditional waterfall model and agile development model, the waterfall model does not adapt to changes in user needs and waste of human resources. The agile model is more adaptable than the waterfall model. It can mobilize everyone’s enthusiasm for work. Faced with long-term construction, frequent personnel changes, and undocumented projects, the impact on the development of the project is more profound, because the handover process is not complete. There are often some unpredictable losses. Take the information management system of a university as an example. A series of operations such as project source code management, testing, and releasing require manual intervention by skilled and relevant personnel. The operation procedures are trivial, and many repetitive tasks cause a lot of waste of manpower, and The operation process is easy to cause the system to crash due to human factors. In addition, the frequent flow of personnel has not formed corresponding standards, and it is very difficult to carry out the follow-up docking personnel work, which has slowed the progress of the new outlook research. Obviously, agile development does greatly improve the speed of software development, but it focuses on the development phased of the software, and does not take the operation and maintenance phase and the necessary collaboration of the team in the process into account.

In order to solve the above problems, DevOps came into being at this time. DevOps integrates development, operation and maintenance, and testing. It expands to the complete life cycle of the software and automates the workflow. It can also eliminate the gap between the development team and the operation team. As new functions under development introduce changes into the system, which increases the risk of downtime, the operations team is not responsible for this. Through the DevOps platform, people can work together to safely make more frequent improvements in a complex environment to avoid making mistake. The development team can devote itself to operations without worrying about the impact of new functions. The DevOps platform guarantees the stability and smooth operation of the system, allowing the operation team to devote itself to operations without worrying about the impact of new functions. The introduction of the DevOps platform has made smaller and smaller batch production feasible, and reduced the cost of testing and deployment to almost zero, so that testing and deployment are essentially free, and small batch work is easier to be understood, tested, reviewed and grasped the time of completion, which can reduce the difference and risk. If there is a problem, you can troubleshoot and recover from it. By combining the automated DevOps platform with solid agile practices, the function development can be very close to the single-piece process, so as to provide customers with value quickly and continuously.

3. DevOps technology selection

The demand for smart campus construction in domestic colleges and universities is changing with each passing day, and the pace of system version iteration is also accelerating. The traditional software model is no longer suitable and cannot help colleges and universities become "first players" in education that overtake and quickly seize the commanding heights of education development.
Therefore, in order to adapt to the development of university education, it is the best choice to build a DevOps platform in the construction of a smart campus. However, considering the existing original management model and system integration and other factors, when selecting the DevOps technology, it is necessary to use the principle of smooth transition and rapid iteration of the system to build a set of efficient and targeted DevOps platform which suitable for colleges and universities.

Automation supported tools already have many mature models: Xebialabs provides a periodic table of DevOps tools, and StackOverdrive provides a panorama of DevOps tools. In academia, Vaasanthi et al. proposed a new method to classify DevOps tools based on data mining technology [7-8], but these models have their own limitations and are not fully applicable to universities. The technical selection of DevOps can be divided into nine modules: agile management tools, product and quality management, code warehouse management, development process specifications, automated build scripts, virtual machines and containerization, continuous integration (CI) and continuous deployment (CD), Automated testing, automated operation and maintenance tools. Different application scenarios can be considered according to their own needs. This article takes the application scenario of campus smart campus as an example, comprehensively considers the management mode of universities, the integration of new and old systems and other factors, to construct a set of DevOps scheme with Jenkins as the core. The technical selection table of the scheme is as follows:

| Module                                   | Common tools/techniques       | Selected tools/techniques |
|------------------------------------------|--------------------------------|----------------------------|
| Agile management tools                   | Trello, Teambition, Worktile, Tower | Trello                     |
| Product and quality management           | Confluence, ZenTao, Jira, Bugzilla | ZenTao                    |
| Code warehouse management                | Git, SVN                       | Git                        |
| Development process specification        | Git Flow, Github Flow, GitLab Flow | Gitlab Flow               |
| Automated build script                   | Gradle, Maven, SBT, ANT        | Maven or Gradle            |
| Virtual machine and containerization     | VMware, VirtualBox, Vagrant, Docker | Vagrant, Docker          |
| Continuous integration and continuous deployment | Codeship, GitLab CI, Travis CI, Bamboo, Jenkins | Jenkins               |
| Automated test                          | Appium, Selenium, Mock Test, Robot Framework, SonarQube | Combine on demand |
| Automated operation and maintenance tools | Ansible, Puppet, Chef, Zabbix, Elastic, ELK Stack | Combine on demand |

The types of agile management tools are complete, with similarities and differences. Software personnel and operation and maintenance personnel who consider breaking the university are often in the same department, and ZenTao also has agile management tools. Compare with other tools, ZenTao is more suitable as a combination of the two selections; In code warehouse management, Git is more secure than a single node service, centralized management version controller SVN, provides intuitive WEB page operation, friendly, and more suitable for multi-project scenarios, its derivative GOGS, Gitlab, Github has more choices, so Git is the preferred choice; the development process is built on Git, Git flows, which is a code of behavior for source code and a tool to simplify some operations, but branch modification is limited and inflexible, and github flow is also one of the tool for version process specification management, focusing on code quality and is not suitable for final product delivery. The GitHub Flow operation is simpler. The code can be directly deployed online by merging feature branches. Considering that smart campus systems is often Final delivery, so GitHub Flow is more suitable; Gradle and Maven are the mainstream automated build scripts, but Gradle is relatively
large and bloated, with many and complex concepts, and high learning costs. Moreover, no matter which automated build script tool you choose, there are manual jar import dependencies based on the traditional legacy system. More importantly, it is easier to upgrade smoothly and choose according to the actual situation; for the release of the built project It is the choice of virtual machine or container. More often, it is the combination of the two, because the virtualization method can meet the user's on-demand resource requirements and ensure the availability and isolation. Docker provides additional isolation methods to avoid virtual operating systems. Occupation; continuous integration (CI) and continuous deployment (CD), recommend Jenkins, which is one of the largest free open source CI tools, has a wealth of plug-ins, can integrate all available tools and services in the market, and can customize your own DevOps The solution helps to automate the deployment; after the deployment, it is necessary to automate the test of the released project to reduce the manpower input of the team test. The main push are the static code quality analysis tool Sonar, which supports 25 languages, can integrate a variety of plug-ins to expand other rules, and can better connect with IDE, Jenkins, Git and other services; automatic operation and maintenance also need to be equipped with monitoring management tools, automation is the operation and maintenance work Sublimation, operation and maintenance automation is not only a maintenance process, but also a management improvement process. It is the highest level of operation and maintenance and is also the future development trend. Common tools commonly used Ansible, Elastic, Puppet, Chef, Zabbix, ELK Stack log analysis system, etc., automated operation and maintenance can be achieved through these tools.

4. DevOps platform construction plan

The technology and software development model used in the construction of the DevOps automated assembly line have problems such as shortage of manpower, high cost of inter-departmental collaboration and communication in the early stage, and large workload and low results. In the initial stage of the project, there is a shortage of personnel and the construction period is tight. Infrastructure construction and personnel communication training are required. The investment cost is high and the results are low. In the construction of a smart campus, system requirements and personnel input are not proportional, and this problem is even more prominent. Aiming at the pains points in the construction of the smart campus, an automated DevOps platform based on Jenkins was built. According to the business flow, multiple tools were effectively integrated, and automated operation and maintenance management was achieved through Zabbix monitoring, data collection, and continuous integration monitoring. DevOps uses automated processes to build, test, and release applications faster, more frequently, and more reliably, thereby improving the efficiency of collaboration between the development team and the operation and maintenance team. DevOps is mainly divided into: planning, cod, construction, testing, version, deployment, operation and maintenance, monitoring and other links, emphasizing teamwork, mutual assistance, and continuous development. However, the traditional model is that developers only care about the development program, and the operation and maintenance are only responsible for basic environment management and code deployment and monitoring. They are not for a common goal to achieve the ultimate goal together, while DevOps achieves team operations, that is, whether it is development, Operation and maintenance or testing, both make their own efforts for the final code release, continuous deployment and business stability, so as to realize the virtuous cycle of product design, development, testing and deployment, and realize the final continuous delivery of the product.

The DevOps platform in this article mainly uses Jenkins, maven, selenium, SonarQube, Elastic, Zabbix, Docker and other tools to build an automated DevOps platform based on Jenkins, through the use of integrated product management, project management, quality management, document management, organization management and transaction management The integrated ZenTao project management software pre-plans the project, uses version management tools such as GitLab to perform version management during the cod phase, and integrates Maven, selenium, shell, SonarQube management tools through Jenkins, etc. to automatically deploy the developed code Test. After the test is passed, the project and environment will be built automatically, the Docker image will be released.
for deployment after the build is completed, the operation will be detected through Elastic, and Zabbix will be monitored. If a problem is found, the repair work is arranged through the ZenTao tool to form a complete DevOps capability loop. The automated process is shown in Figure 1.

Figure 1. Automation flow

5. DevOps platform experiment

Through the EXSI platform, a 4-core 16G Centos was created, and a DevOps platform based on Centos7 was built. The platform mainly selected some tools suitable for college DevOps, the core ones are Jenkins, maven, selenium, SonarQube, Zabbix, Docker, etc. On the platform, the smart welcome project (evolved from the legacy system of the new welcome) in the construction of the smart campus is placed on the platform for comparative testing, comparing the traditional development process (waterfall mode or agile mode) and the development after the platform is introduced. The process takes time to complete a complete product iteration.

The main process of DevOps are to build a Maven project through Jenkins, configure it to pull code from git, and use the configuration hook to identify when someone submits the code to automatically release the build or manually trigger the automatic built operation. Start the relevant packaging program through the script, such as starting the Node environment to automatically package the front-end code to generate a jars package, and then hand it over to Maven to build the back-end code and merge to generate the project jar package, and then generate the running environment of the project through Docker-compose and automatically start the project, And finally released to SonarQube to review the project code.

A complete product iteration requires 8 steps: demand management, product definition, interactive design, technical review, research and development, testing, launch, and tracking optimization. Under the traditional development model, it takes about one month for the product to iterate the welcome system. In the more efficient agile model, the time required for the early stage from demand management to technical review and later R&D to tracking optimization is about half a month. Among them, more time is spent in the early stage on demand management and product definition. In this process, more time is spent on communication, and most of the later time is spent on version release. Each release of a version requires operation and maintenance personnel to pull the code for manual deployment. If the developer makes changes to the project environment, Operation and maintenance personnel also need to manually adjust the project environment.

After the introduction of the DevOps platform, first of all, the preliminary work are carried out on Zen Tao, where developers and testers participate, and in the process, the "story mode" inside is used to form a testable requirement document, so that the test can be tested as early as possible. The project team communicates in ZenTao, and can consult documents and iterative versions related to each requirement at any time, which speeds up the efficiency of communication. Furthermore, the later
work can rely on the automated construction platform with Jenkins as the core. Through this platform, you can save development iteration costs from three aspects. First, you can set coding specifications in advance, and developers can use the platform to review and modify their own code to reduce bugs caused by specifications; second, testers can advance Entering test cases allows the platform to automate testing, saving manual testing time and cost; third, the operation and maintenance personnel write the release script, and then the version release only needs to modify the release script when the environment changes. Iterating the above products once through this platform section can save about 1/5 of the time.

6. Summary
Through the use of the DevOps platform, in the process of building a smart campus in universities, efficient delivery, rapid iteration, automated operation and maintenance, reduce the frequency of manual participation, and avoid falling into trivial and error-prone blind spots. Through the practical use of the platform, compared with the traditional development and operation and maintenance methods, the development iteration times is greatly shortened, and it provides a reference for colleges and universities to build the DevOps platform in the construction of smart campuses.

Acknowledgments
This article is a major scientific research project in Guangdong Province, "Research on System Optimization for High Concurrency and Complex Course Selection Business" (2018KTSCX340) and the 2020 South China Institute of Software Engineering GU Scientific Research, Education and Teaching Research Project "Research on Automated DevOps Platform Based on Jenkins and Docker" (ky202023) is one of the phased achievements.

References
[1] Gan, R., Yuan, Z., He, G. (2019) The latest development and enlightenment of foreign smart campus construction[J]. Modern Educational Technology, 29(02):19-25.
[2] Li, W., Yuan, X., Elhoseny, M.. (2021) Design of smart campus management system based on internet of things technology[J]. Journal of Intelligent & Fuzzy Systems, 40(2): 3159-3168.
[3] Liu, B., Zhang, H., Dong, L. (2019) Research on DevOps in China[J]. Journal of Software, 30(10): 3206-3226.
[4] Jabbari, R., Ali, N., Petersen, K., Tanveer, B. (2016) What Is DevOps? A systematic mapping study on definitions and practices. In: Proc. of the Scientific Workshop of XP. Edinburgh, 1–11.
[5] Lin, P., Shi, X., Yang, J., Yang, S., Lin, Z. (2020) Performance and security optimization of large legacy systems[J]. Computer System Applications, 29(10): 103-108.
[6] An, J., (2017) Thinking and practice of the construction of a new smart campus[J]. Beijing Education (Higher Education), 2017(11): 6-9.
[7] Huang, H., Zhang, H., Shao, D. (2019) The impact of automation tools on DevOps practice in China [J]. Journal of Software, 30(10): 3056-3070.
[8] Vaasanthi, R., Prasanna, V., Philip, S. (2017) Analysis of DevOps tools using the traditional data mining techniques. Int’l Journal of Computer Applications, 161(11):47–49