Design and Implementation of an Information Management System for College Students in Higher Education Institutions Based on Cloud Computing

Mo Bin
Guangdong Polytechnic of Science and Technology
Zhuhai 519090, China

***Abstract*—A cloud computing-based system has been developed to enhance the efficiency and practicality of the information management system for college students in higher vocational colleges. This system incorporates a well-defined architecture that leverages cloud computing technology. The management layer's logic module ensures the security of vocational college students' information by deploying virtual gateways at strategic points within the system, thereby controlling access, sharing, and exchange of information. The resource module in the application layer optimizes server cluster load balancing by minimizing task completion time and improving load balancing effectiveness. Additionally, the M-Cloud storage mode is employed to store and back up application layer cloud information, along with the distributed Bigtable information base. The user access layer provides users with convenient services through the corresponding cloud service access interface in the application layer. Furthermore, the employment information of college students and enterprise position information are clustered using the K-means algorithm based on data mining, and personalized employment recommendations are made using similarity calculations. Experimental results demonstrate that the system boasts a user-friendly interface design, efficient operation, and comprehensive management functions. The system's server cluster exhibits strong load-balancing capabilities, effectively mitigating network congestion and minimizing the risks of network storms and paralysis.**

Keywords—Cloud computing; student information; management system; load balancing; virtual gateway; personalized recommendation

I. INTRODUCTION

As an important component of the national education system, vocational colleges face enormous challenges in student information management due to their large number and diverse types of students. Automated management of student information in higher vocational colleges and universities through the information management system can reduce manual operation, reduce the error rate, and improve management efficiency; the system can be encrypted and backed up to ensure the security and integrity of the data [1], through the system can be convenient to share the student information, to promote the collaboration and communication between the departments [2], the data mining and analysis, to provide decision-making support for the leadership of the school to improve the management level and competitiveness, but also to provide more convenient services for students, such as grade inquiry, course scheduling [3], to enhance student satisfaction and teaching quality.

There are many methods for the design and implementation of information management systems at home and abroad. For example, Li D C et al. [4] proposed a new SF deployment management platform, which aims to achieve the dynamic deployment of edge computing service applications with the lowest network latency and service deployment cost in edge computing network environment, and verify its practicability in pure edge computing and mixed edge cloud computing scenarios through experiments. This method also proposes a solution to the problem of network load balancing, but the comprehensive load capacity of the server is not fully considered, resulting in long network response time and easy occurrence of network unresponsiveness. American K et al. [5] proposed a semi automated method for ensuring network physical security, which can automatically identify and verify network security related statements in industrial control system equipment documents through the development of new algorithms and tools, thereby assisting in the generation of compliance reports and reducing OT device security risks. However, this method does not take into account the issue of storing a large amount of data in the system, which may lead to server overload. Bi D et al. proposed the Internet of Things (IoT) assisted college students' information management system using hybrid crypto-integrated steganography technology [6]. This paper uses a hybrid crypto-integrated steganography (HCIS) algorithm and auxiliary data input for college students' information management systems. It uses the Internet of Things to help secure data sharing in the cloud environment. The code generated by password students can provide a high degree of privacy for users accessing cloud data. Cryptography converts data into a safe format readable by authorized users, and steganography helps to transmit secret data to avoid information discovery and uses encryption keys to hide or effectively protect data, thus realizing the design of college students' information management system. This system ensures the security of student information but does not consider the problem of network access load balancing, which may lead to the risk of network paralysis. Chen W et al. proposed a dynamic student information management system based on computer vision for higher education platforms [7]. This paper introduces a dynamic student data management system (DSDM-AICV) based on artificial intelligence computer vision technology. Based on the collected information, use AI-enabled archiving and dynamic user access to generate data-related process flows, which helps to explore the relationship between student data, improve the level of data management, and achieve student information management. This method does not recommend employing college students, so the system needs to be improved.

The cloud computing platform offers flexible resource allocation and expansion capabilities, allowing for dynamic resource adjustments based on system requirements at various stages. It incorporates a comprehensive security mechanism, encompassing data encryption, access control, and more, to ensure the confidentiality and security of student information. Cloud computing platforms have high availability and fault tolerance, which can ensure system stability and reliability, avoid system crashes or data loss caused by hardware failures or network issues, and provide efficient computing and storage capabilities. They can quickly process large amounts of student information data, improve management efficiency and service quality. Therefore, this article adopts a cloud computing platform to design and implement an information management system for college students in vocational colleges, achieving centralized storage, efficient management, and secure sharing of student information, which has become an urgent need for information construction in vocational colleges. This new management method not only improves the efficiency and quality of student information management in vocational colleges, but also provides more accurate and comprehensive data support for educational decision-making in schools.

The specific implementation process of this paper is: first, build the student information management system architecture, and optimize the server cluster load balancing to improve the system performance. Then, the M-Cloud storage mode is used for cloud information storage and backup, and the system security is guaranteed through the virtual gateway. Finally, the data clustering algorithm is used to realize the personalized employment recommendation for college students, complete the system design, and provide efficient and stable student information management solutions for higher vocational colleges.

II. Design of Information Management System for College Students in Higher Education Institutions

A. *Information Management System Architecture for College Students in Higher Vocational Colleges and Universities Based on Cloud Computing*

The architecture of the information management system for college students in higher vocational colleges and universities, based on cloud computing, consists of three main components: the management layer, application layer, and user access layer. The management layer encompasses the gateway, logic, and student network modules, which oversee all levels of cloud computing services [8]. On the other hand, the application layer comprises the resource module, platform module, and application module. The resource module encompasses physical resources, server services, storage services, and network services, while the platform module includes database services and middleware services [9]. Lastly, the application module encompasses front-end application services and student information management application services. The core aspect of cloud computing in the architecture of information management systems for college students lies in providing services through the Internet. Consequently, the resource, platform, and application modules offer infrastructure, platform, and software services, respectively. The user access layer, also known as the user access layer, primarily includes a service directory, subscription management, service access, and personalized recommendations.

Fig. 1 illustrates the structure of the information management system designed for college students in higher education institutions, which operates on cloud computing technology.

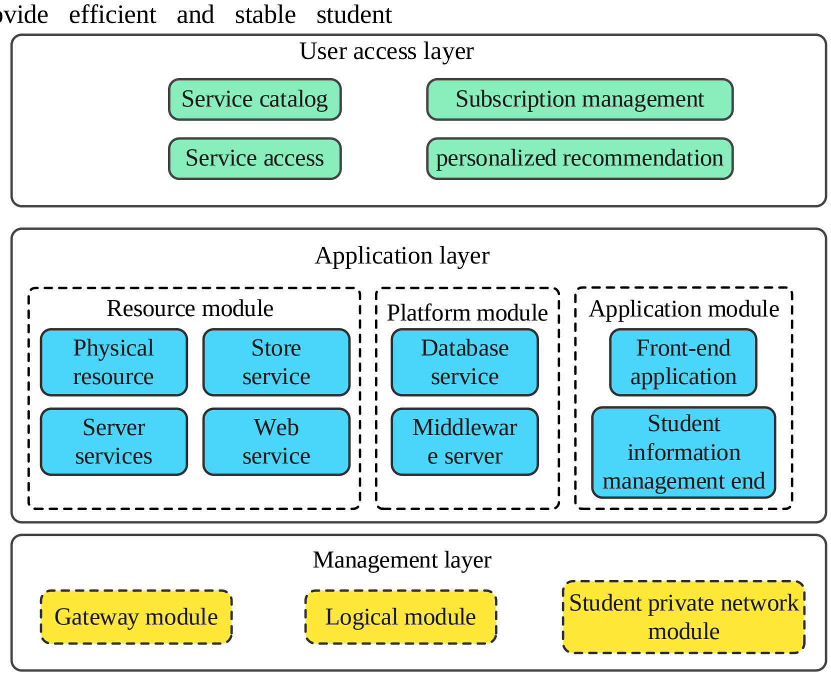


Fig. 1. Architecture of college student information management system based on cloud computing.

The management layer, serving as the central component of the college student information management system in higher vocational colleges and universities, encompasses various modules such as the student's particular network module, logic module, and gateway module. These modules collectively enable the implementation of student information security control within the system.

The resource module in the application layer refers to providing the infrastructure resources of cloud computing as a service to users. Users can build their applications on these essential services. This service conceals complex physical resources from users by providing virtualized resources. Physical resources refer to many physical facilities supporting various services on the upper layer of cloud computing, including network equipment, servers, and storage devices [10]. Among them, server services can provide server environments, including Linux, Unix, Windows, and even server clusters, with the support of virtualization technology. Storage service can provide storage function. General network processing functions provided by network services, namely VLAN, load balancing, route switching, firewall, etc.

The platform module is an abstract encapsulation of the resource module services; after processing the large granularity of the resource module, services are more advanced and easy for users to create their applications [11]. Middleware services provide users with scalable transaction middleware or messaging middleware, and database services provide users with scalable database processing capabilities.

The application module covers all the software methods for system operation, which is built on top of the base layer and data layer to provide different application services to different objects.

The user access layer provides various convenient services for users of cloud computing services and provides corresponding cloud service access interfaces for each level as required. Users can select the necessary cloud computing services from the service list in the service directory. It provides the access interface of the resource layer for the remote desktop and the access interface of the application layer for the Web [12]. The subscription management function manages customer information or terminates customer subscriptions.

B. Management Design

The logic module in the management layer is the critical module in this layer, which realizes the integration, storage, sharing, and interaction of college students' information, and the gateway module includes the integration of virtual gateway, firewall data encryption decryption, etc. The structure of the management layer is shown in Fig. 2.

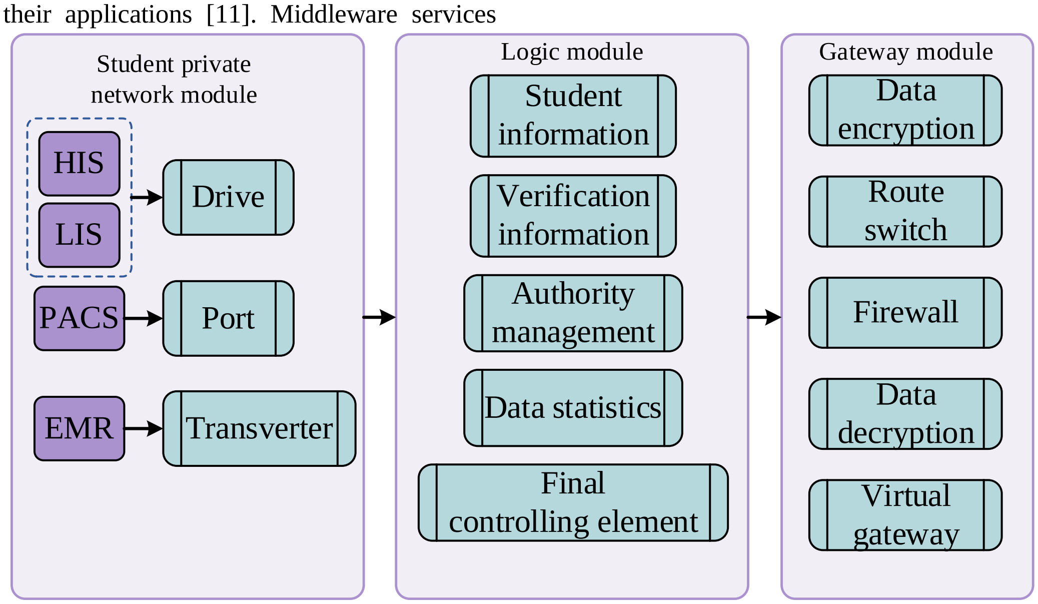


Fig. 2. Management structure.

Logic module according to the service demand, the college students need to share information or interactive information through the security gateway, and complete information communication; At the same time, the module can complete the verification of the basic information related to college students' data information, and support the access of all mobile terminals [13], after the completion of the identity verification can be carried out within the corresponding permissions of the operation, to achieve the interaction of the basic data information of the students and the sharing of the information.

Users need to access the relevant applications in the system through the virtual gateway. Therefore, the management and application of primary student information data is accomplished under the control of the virtual gateway of the information management system, which is controlled by the virtual access domain Y access to that domain intranet N, and the implementation of the relevant information operations is realized. In this process, the information must pass through three trust domains to discover the final purpose. Using a virtual security gateway to realize information security control, the gateway will be deployed in the key nodes. This paper mainly uses nine kinds of virtual gateways to realize the security control of college students' information management in higher vocational colleges and universities.

Virtual Gateway 1 and 2: The gateway has secure terminal connection access control. It is mainly used to ensure secure access to users and the security of logical boundaries, and both belong to Y.

Virtual Gateway 3: The gateway mainly guarantees the access security of the Internet H, and is connected by N, and has the port forwarding function.

Virtual Gateway 4: This gateway is used to realize the security of information interaction among college students, and it belongs to the interaction between N and Internet computing domain J. The gateway has the function of intrusion detection.

Virtual Gateway 5: The gateway is also used to realize the safe interaction of information, and the information belongs between N and the public storage domain, and the gateway has the function of adding and decrypting information.

Virtual Gateway 6: This gateway is used to realize the operational security of the security services intranet, which can transport remote access to realize remote security connections.

Virtual Gateway 7: This gateway is used to secure the connection between the security administrator of the student information management system and the system security service.

Virtual Gateway 8: This gateway is used for security control of cloud storage access functions.

Virtual Gateway 9: This gateway can be understood as a virtual host of the firewall; all the information sent to the virtual host needs to be protected through the gateway before the implementation of the continued operation; the gateway has an application layer of intrusion, attack, and other defense and detection.

To summarize, the virtual gateway is deployed in each vital position of the system, and all the access, sharing, exchange, and other operations of the information are controlled by the virtual gateway [14], which ensures the security of the college students' information in all the links to guarantee the security of the operation of the college students' information management system.

C. Application Layer Design

1) Load balancing optimized scheduling of server clusters in the resource module: When the server cluster handles requests for college students' information management tasks in higher vocational colleges and universities, problems such as unbalanced distribution of requests and long task completion time will occur [15], so load balancing needs to be optimized for scheduling.

The task completion time for a single server in the cluster is the time elapsed from the start of a college student information management task request to the completion of task processing, the waiting time of task T_i in the queue is $t_{wait}^{(T_i)}$, the actual processing time is $t_{deal}^{(T_i)}$. The waiting time for information management task requests for college students in higher vocational colleges and universities is generally determined by the current network conditions and the size of the real-time load in the cluster, which is randomized. In comparison, the actual processing time of information management tasks for college students in higher vocational colleges and universities is determined by the amount of tasks that $m^{(T_i)}$, the combined processing power of the matching servers handling the request $Q^{(T_i)}(U_j)$ and aggregate load information for that server at the time of the request $A^{(T_i)}(U_j)$. Usually, the processing time of a task request with no other load on the server is determined by $t_{emp}^{(T_i)}$, the formula given by Formula (1).

$$t_{emp}^{(T_i)} = \frac{m^{(T_i)}}{Q^{(T_i)}(U_j)} \quad (1)$$

Among them, $m^{(T_i)}$ indicates the volume of tasks requested by the mandate and measures the task's degree of difficulty. $Q^{(T_i)}(U_j)$ represents the combined processing power of the matching server, and the ratio of the two represents the running time of the task request time T_i on the matching server U_j without load. Generally speaking, the purpose of using clustering to deal with college students' information management task requests is to solve the problem of insufficient processing capacity of a single server, applied in the concurrency scenario [16], mainly to test the concurrent processing ability of the server, so there is rarely a server in no load for a single task processing, usually at the current moment when the server has a load task for the following task processing [17], i.e., matching the situation where the server is loaded. At present, the relationship between the processing time of the information management task request T_i under the matching server U_j and the processing time under no load is shown in Formula (2):

$$t_{deal}^{(T_i)} = t_{emp}^{(T_i)} * \frac{1}{1-A^{(T_i)}(U_j)} \quad (2)$$

Among them, $t_{deal}^{(T_i)}$ represents the task processing time when there is a load, calculated from the unloaded task processing time $t_{emp}^{(T_i)}$ and the real-time load information $A^{(T_i)}(U_j)$ of the matching server when the task request arrives. The comprehensive load information for each server is evaluated through four resources, including CPU usage, memory usage, network bandwidth usage, and disk I/O usage, the combined load information of the A of server U_j can be expressed as Formula (3) through a linear formula.

$$A(U_j) = \alpha_1 * A_{CPU}^{(U_j)} + \alpha_2 * A_{memory}^{(U_j)} + \alpha_3 * A_{band}^{(U_j)} + \alpha_4 * A_{I/O}^{(U_j)} \quad (3)$$

Among them, $A_{CPU}^{(U_j)}$ denotes the CPU utilization rate of the server U_j, others are the same; α_i is weighting coefficients indicating various resource occupancy rates, which may be set according to the actual processing of transactions, but which satisfy $\sum_i^4 \alpha_i = 1$and $A(U_j) \in (0,1)$.

Combining Formula (1), Formula (2), and Formula (3), when a request for an information management task for college students in higher vocational colleges and universities arises, a reasonable selection of matching servers to execute the task is the key to solving the problem. The appropriate matching server is selected based on the size of the server's real-time load information $A(U_j)$. Therefore, the minimum allocation ratio threshold is introduced to determine the task matching server, expressed by Formula (4).

$$0 \leq \frac{A(U_j)}{\sum A(U_j)} \leq thr_{A(U_j)} \quad (4)$$

Among them, $\frac{A(U_j)}{\sum A(U_j)}$ represents the ratio of the time of load information of the server U_j at the arrival of the task request to the total load information in the cluster at that time. $thr_{A(U_j)}$ denotes the minimum assignable threshold, when $\frac{A(U_j)}{\sum A(U_j)}$ is less or equal to this threshold, it means that this server node is lightly loaded and can accept new task requests. The smaller the $\frac{A(U_j)}{\sum A(U_j)}$ is, the newer requests the server can accept. When the request arrives, if there exists more than one server node to satisfy the above formula, a merit selection will be made, i.e., the server node with the smallest ratio will be selected for the processing of the current task request of information management for college students in higher vocational colleges and universities. When a group of task requests are assigned, the size A_{on} of the number of task connections of each server is recorded while recording the relationship $B_{T_i}^{m}$ of each task request to the matching server. $B_{T_i}^{m}$ is a row vector representing a row in the matching set, e.g., 12 tasks are being processed on 7 servers, where the 3rd task is being processed on the 5th server, then, the $B_{T_3}^{m} = [0000100]_{1\times 7}$, which are inserted into the matching set $B = \left[B_{T_1}^{m}, B_{T_2}^{m}, B_{T_3}^{m}, \cdots, B_{T_{11}}^{m}, B_{T_{12}}^{m}\right]_{12\times 7}^{T}$ in task order after the request is completed.

Synthesizing Formula (1) ~Formula (4), the completion time of all college students' information management task requests on the server is represented by a multivariate combination, and the expression is Formula (5).

$$t(U_j) = \left[t_{deal}^{(T_i)}, t_{wait}^{(T_i)}, B_{T_i}^{m}\right] \quad (5)$$

Formula (5) can be transformed into Formula (6).

$$t(U_j) = \left[t_{deal}, t_{wait}, A(U_j|T_i), Q(U_j|T_i), A_{on}, B\right] \quad (6)$$

Among them, $t(U_j)$ indicates the time for a single server in the cluster to complete all tasks. t_{deal} and t_{wait} are the sum of the processing time and the sum of the waiting time for all tasks on that server, respectively. $A(U_j|T_i)$, $Q(U_j|T_i)$indicate that the task request T_i matches the real-time load information and comprehensive processing capability of the server U_j when the task request arrives; A_{on} denotes the set of connections. B denotes the set of task matches. Under the premise of considering the server's real-time load information of the serverand the comprehensive processing capacity, the task completion time of a single server is the sum of the processing time and waiting time of each server task. The task completion time t_a in the whole cluster can be written as Formula (7).

$$t_a = \sum_{j=1}^{n} t(U_j) \quad (7)$$

At the same time, the load balancing validity degree δ is introduced to measure the difference in the processing time of information management tasks of higher vocational college students among the servers in the cluster, which is expressed by Formula (8) as follows:

$$\delta = \left[\sum_{j=1}^{n}\left(t(U_j) - t_{avg}\right)^2 / n\right]^{\frac{1}{2}} \quad (8)$$

Among them, $t_{avg} = \frac{t_a}{n}$ denotes the average task completion time for each server in the cluster; the δ denotes the variance of the processing time of each server in the cluster, the smaller the value δ, the more balanced the processing time of the servers in the cluster, and the better the overall performance of the servers.

Minimizing the task completion time and enhancing the load balancing effectiveness of the cluster are two objectives of the load balancing problem in clusters [18]. Theoretically speaking, these two objectives are competing with each other, to keep the cluster in a relatively balanced state, to ensure that the processing time of each server does not vary much [19], it is inevitable to sacrifice the task completion time of individual servers at the expense of the task completion time of the entire cluster, so the multi-objective model proposed in this paper is established as expressed in Formula (9).

$$\begin{cases} min\ t_a \\ min\ \delta \\ s.t. \quad \sum_{j=1}^{n} b_{i*j} = 1 \\ thr_{A(U_j)} \leq a \in (0,1) \end{cases} \quad (9)$$

The 1st constraint in Formula (9) specifies that each task is executed on one server and only on one server; the 2nd constraint specifies that the maximum allocation ratio must satisfy a constant between 0 and 1.

The above can make the server cluster to keep the load balanced when dealing with the request for information management of college students in higher vocational colleges and universities.

2) Storage services in the resource module

a) M-Cloud storage mode

The storage service in the resource module of the information management system application layer for college students in higher vocational colleges uses M-Cloud storage mode to realize the storage and backup of cloud information for college students in the application layer. M-Cloud storage mode is an improved key-value storage management mechanism, which uses three methods of segmentation, buffering, and addition to achieve effective storage and management of college students' data in higher vocational colleges [20]. The specific composition of the M-Cloud storage mode is shown in Fig. 3.

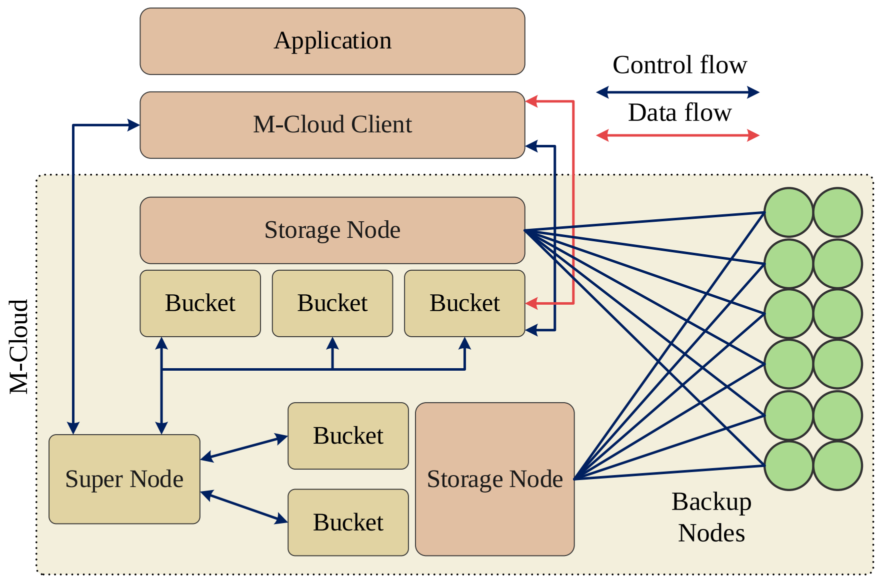


Fig. 3. M-Cloud Storage mode structure.

As shown in Fig. 3, the M-Cloud storage mode consists of multiple storage servers (StorageNodes), several buckets, a SuperNode, and multiple backup servers (BackupNodes). The super server manages the cloud information of vocational college students in the whole storage server and manages the cloud information activity range under M-Cloud storage mode. To obtain real-time status information and prevent cloud data loss, regular communication between the super server and each storage server is required [21].

The storage server manages all cloud information collections under M-Cloud storage mode and is used for operations such as reading, deleting, and backing up information. The super server allocates information to each storage server for management, reducing the burden of the super server, and improving the storage speed of cloud information.

Use the bucket concept to avoid the shortcomings of performance differences among storage servers, where the storage capacity of each bucket is consistent. M-Cloud storage mode uses buckets to achieve mass information exchange among vocational college students [22]. Configure the bucket and storage server in multiple to single configurations to achieve weight configuration and load balancing.

Multiple backup servers are used to back up college students' information stored in different storage servers on disks, which can be backed up and restored when the storage servers fail.

b) Distributed bigtable information base

Transfer the information of vocational college students stored in M-Cloud storage mode in the resource module to the distributed Bigtable information database, further layout and save the information stored for vocational college students, and improve the information throughput of massive vocational college students. The distributed Bigtable information base is composed of three important components: the master server, the client-connected library (ClientLib), and the table server (TabletServer), as well as the ORACLE lock and GFS file server. As shown in Fig. 4.

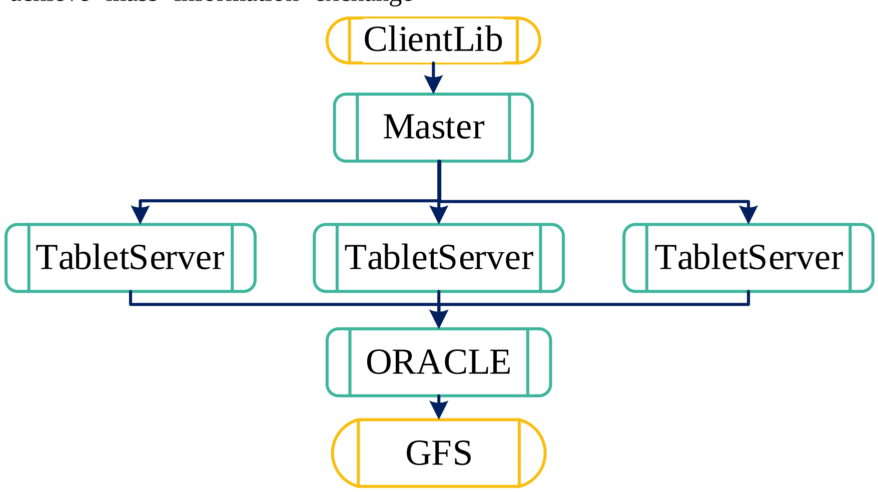


Fig. 4. Distributed bigtable database structure diagram.

Create a unified interface in the database connected to the user end, and transmit the information stored for college students in higher vocational colleges to the main server, responsible for distributing the stored information evenly to each table server. Each table server carries out a distributed information layout for the stored information, locks the managed information with ORACLE, and saves the locked information with the GFS file server [23].

The Bigtable information base is used to distribute and save massive stored information, improve the throughput of massive information, and avoid the overload of cloud computing servers.

3) Application modules: The structure of the application module is shown in Fig. 5. The application module consists of two parts: the front-end application and the student informationization management end, and the front-end application includes the statistical analysis of student information data, study plan, business management of student information, and query of student information; the cloud resource management of student information, student information management, business management and so on form the management end of student informationization, among which, the cloud resource management of students completes the scheduling and supervision of student information and resources through relevant methods; the platform management and business management are used to complete the basic data management, terminal management and business configuration, docking management and statistical analysis.

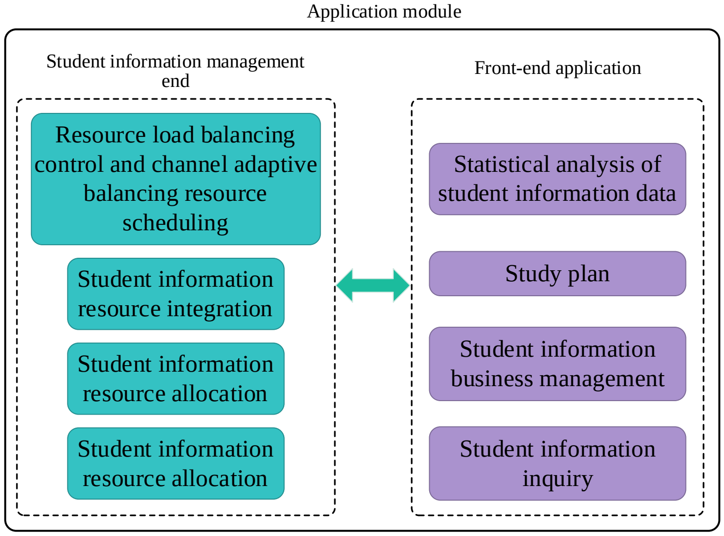


Fig. 5. Application module structure.

D. *Personalized Recommendation Model of College Students' Employment Information Based on the K-Means Algorithm in the user Access Layer*

Based on the data collected from college students in higher vocational colleges, this study examines the current and potential individualized requirements of college students regarding employment. It also investigates the methods and patterns through which college students acquire employment-related information, and establishes a personalized database. The utilization of cloud computing technology offers substantial computational and analytical capabilities, as well as scalability, making it suitable for handling extensive datasets. Consequently, in the deployment environment of cloud computing, the K-means algorithm can be swiftly implemented. This algorithm, rooted in data mining, is the technology for analyzing and processing college students' employment information and enterprise position data. Using the clustering outcomes, a personalized recommendation model for college students' employment information is developed, employing data mining techniques. The structure of this model is depicted in Fig. 6.

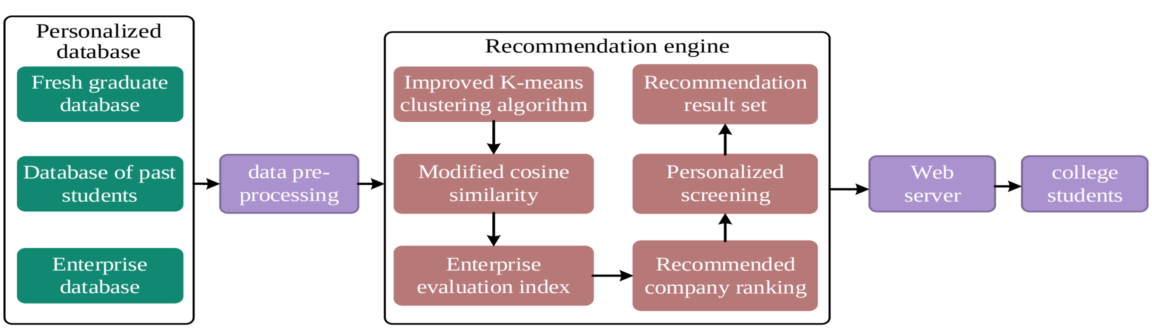


Fig. 6. Personalized recommendation model of college students' employment information.

Through the reality and potential employment needs of college students, establish a personalized database and determine the data range of college students' information and enterprise position information to meet the target of college students' employment information recommendation. In building a personalized database, it is necessary to design a customized data structure to ensure the integrity of each piece of information as far as possible. To achieve targeted recommendation services for college students' employment information, the K-means clustering algorithm in data mining can be used to mine hidden valuable information in the personalized database, accurately discover the knowledge in the personalized data [24], complete information classification according to the natural attributes and use characteristics of information and other variables, and improve the personalized recommendation effect of college students' employment information.

Use data cleaning, integration and attribute specification to preprocess the data of the current student database, previous student database and enterprise database [25]; By improving K-means clustering algorithm, the employment information of college students in the database of preprocessed new students and previous students is clustered, and the enterprise position information in the enterprise database is clustered; according to the clustering results, the modified cosine similarity method is used to calculate the similarity between new students and previous students, as well as the similarity between new students and enterprises; the empirical formula is used to solve the enterprise evaluation index, which is used to express the overall strength of the enterprise; According to the similarity between new graduates and enterprises, as well as the enterprise evaluation index, the enterprise ranking weight is obtained; Consider the factors such as the workplace and salary that college students care about, conduct personalized screening of enterprises, and obtain enterprise rankings; Personalized recommendation results for college students' employment information based on enterprises ranking top.

The clustering of college students' employment information and enterprise position information uses the K-means clustering algorithm to cluster and process the current student database, previous student database, and enterprise database to obtain the clustering results of current and prior students' employment information and enterprise position information. The principle of the K-means clustering algorithm is: first obtain k student the employment information (enterprise position information) object x, treating it as the clustering center of the cluster C; and then solve the distance d between the object x and C of the initial k of the student employment information (enterprise position information); Move x to the class in which the C of the minimum d is located; and then solving for the mean of all the samples of student employment information (enterprise position information) in the cluster, obtaining the new C, marked as C_{new} Finally, it is iterated repeatedly to complete the convergence until the output of the clustering results of the student employment information (enterprise position information). The clustering criterion function of the algorithm is shown in Eq. (10).

$$L = \sum_{i=1}^{k} \sum_{p \in C_i} \|x - M_i\|^2 \quad (10)$$

Among them, the cluster center of the i th student employment information (enterprise position information) object is C_i; The mean value of C_i is M_i. The role of L is to acquire the k interclass dissimilarity and intraclass proximity of individual clusters are elevated.

The process of K-means clustering algorithm clustering college students' employment information (enterprise position information) is shown in Fig. 7.

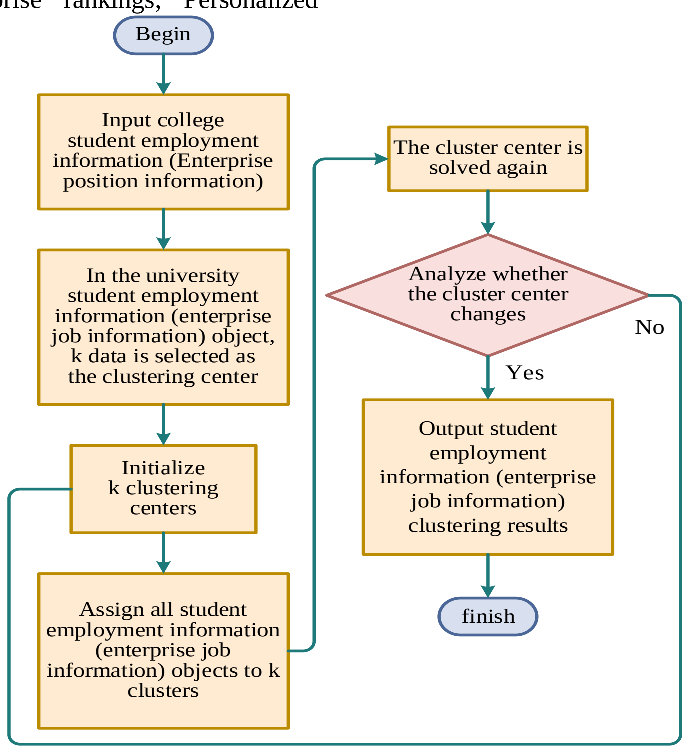


Fig. 7. Clustering process of college student employment information (enterprise job information).

E. Data Flow Diagram of Information Management System for College Students in Higher Education Institutions

A data flow diagram (DFD) serves as a visual representation of the system's logical function, the direction of logical flow, and the logical transformation of data within the student information management system. It is a primary tool for expressing structural system analysis methods and employs graphical techniques to depict software models [26]. The accompanying data flow diagram does not include specific physical elements; instead, it solely describes the system's flow and processing of information. Based on the current business process of managing college students' information, the data flow diagram is constructed by initially identifying college students as the source and endpoints and refining them to obtain the data flow diagram after outlining the logical system, as illustrated in Fig. 8.

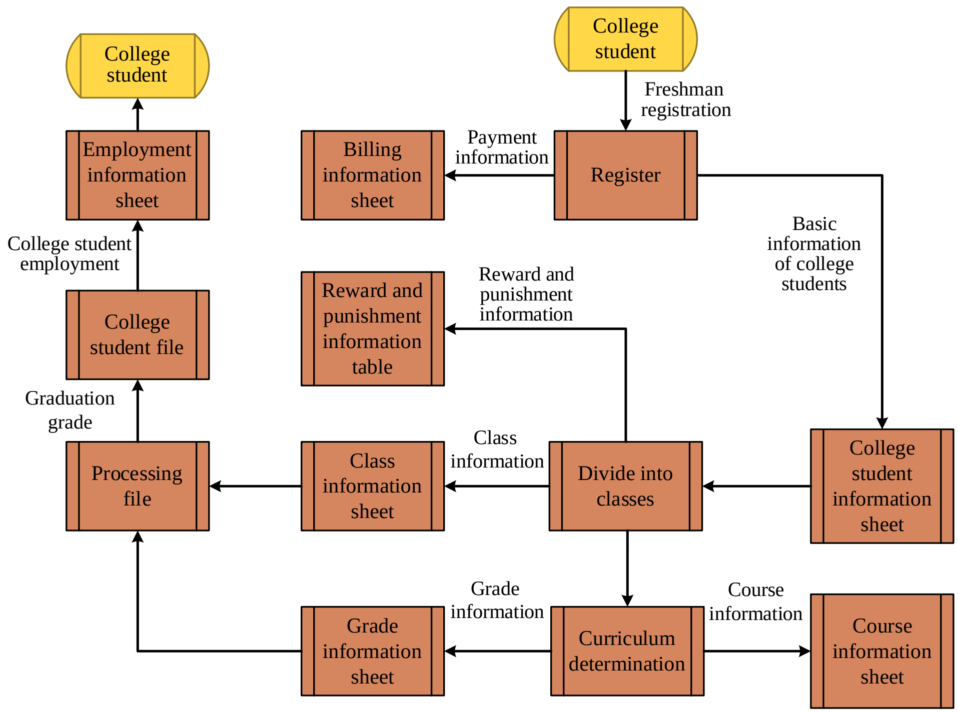


Fig. 8. Data flow diagram of college student information management system.

The beginning and the end point of the data flow in Fig. 8 are both college students; after registration for the new students, students' payment information is stored in the payment information form. The student's basic information is stored in the student information sheet. Each department is divided into classes according to the majors the students admitted, and the class classification results are stored in the class information sheet. Students enter a normal college study and life, and after being rewarded and punished, information is stored in the reward and punishment information table. The course information sheet records relevant details Upon establishing the course. Subsequently, the final examination is administered, and the outcomes are documented in the score information sheet. The student's particulars are then filed, signifying their graduation information. Furthermore, when a student secures employment, the pertinent job details within the enterprise are stored in the employment information table. Consequently, developing an information management system for higher vocational college students is accomplished.

III. Experimental Analyses

A. Experimental Verification

A vocational and technical college is selected to apply the methodology of this paper to design a management system for its college students' information. This vocational and technical college has over 14,000 full-time students and 718 teaching staff. The school has 10 secondary colleges and 48 majors (including directions) in 15 categories. The server cluster system consists of a load balancer in the realm, three real servers in the backend, and a database server, whose configuration is shown in Table I.

TABLE I. SERVER CLUSTER SYSTEM CONFIGURATION

Name	CPU	Operating system	Internal storage	Hard disk
Database master server	Intel Core i7-8700K 3.7GHz	Ubuntu 14.06	4GB	480GB
Load balancer	Intel Pentium(R) 4 30.6GHz	Ubuntu 14.06	8GB	512GB
Slave server	Intel Core i7-8700K 2.8GHz	Ubuntu 14.06	4GB	480GB
Slave server	Intel Pentium(R) 4 30.6GHz	Ubuntu 14.06	4GB	480GB
Slave server	Intel Core i7-8700K 4.2GHz	Ubuntu 12.04	8GB	512GB

For freshmen, the information management system is designed, and the system's functional modules are implemented. The operating interface of the student information management system is shown in Fig. 9.

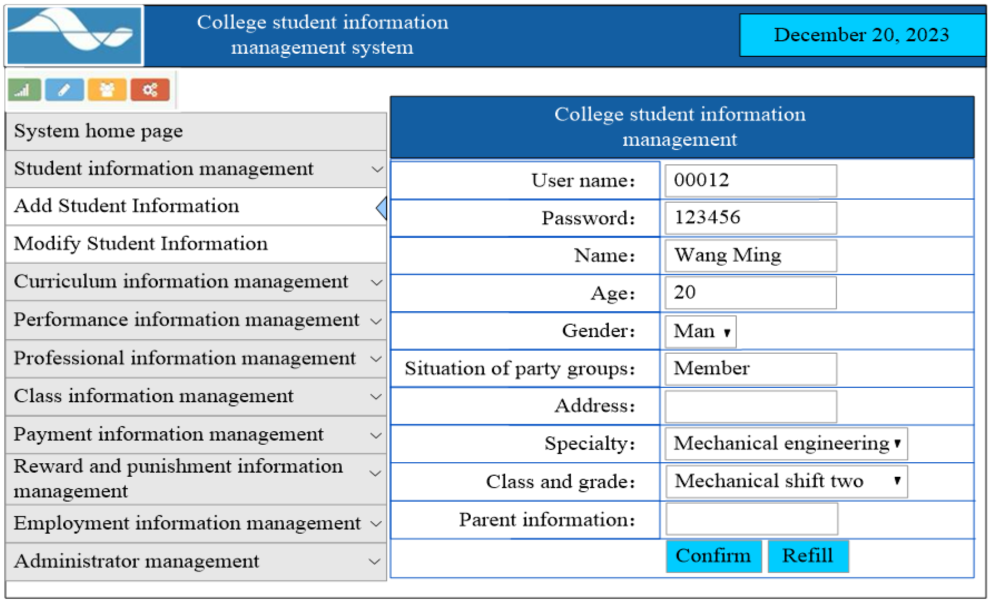


Fig. 9. Student information management system operating interface.

As can be seen from the interface in Fig. 9, users will enter an interactive interface with various menus after logging into the student information management system. These menus correspond to different functional modules, providing convenient and intuitive operation methods for administrators and users. Multiple menu items will be displayed on the main interface when the administrator successfully logs in to the system. These menus include: system menu, student information management menu, course information management menu, achievement information management menu, professional information management menu, class information management menu, payment information management menu, reward and punishment information management menu, employment information management menu, administrator management menu and so on. Click each menu, and the system will enter the corresponding sub-interface at the next level, providing specific operation options and functions. For example, click "Student Information Management Menu," and the system will enter the sub-interfaces to add student information, view student information, and modify student information. Each sub-interface has straightforward tips and operation steps so administrators can easily carry out various operations. The current operating interface is the new student registration, adding the student information interface. The interface layout is straightforward, and the operation steps are clear. The user only needs to input the necessary information according to the prompt, and the system will automatically add the information to the database without complicated operation. The student information management system has a clear and easy-to-understand interface design, convenient and quick operation mode, and comprehensive and rich management functions. Both administrators and ordinary users can easily get started and quickly master the use of the system.

Take Wang Ming, Mechanical Class II, for example, to query its course information, as shown in Fig. 10.

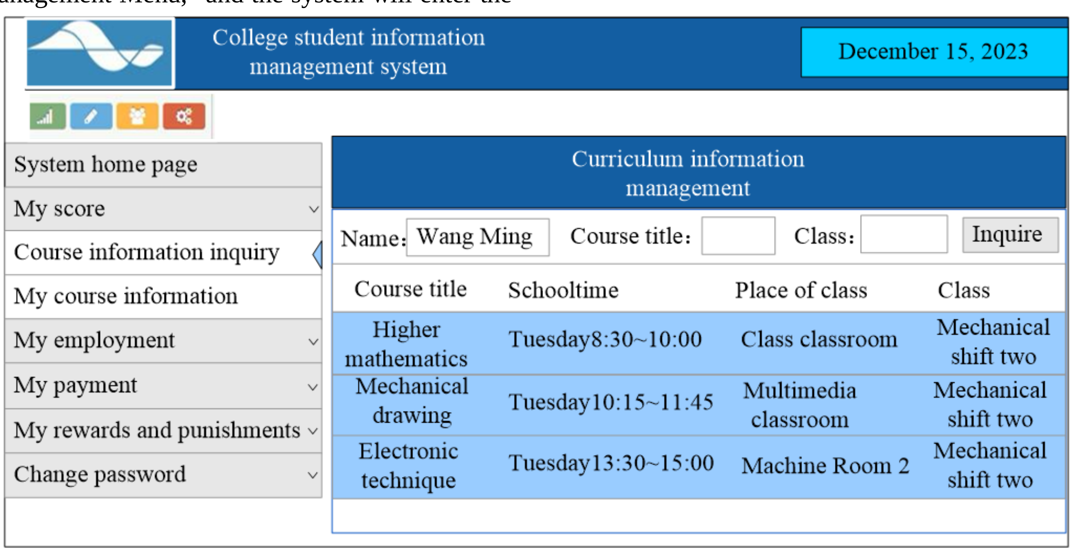


Fig. 10. Wang ming's query of course information.

As seen in Fig. 10, when you click the course information management menu in the main interface of the operating system, you will enter the course information interface. In this interface, several menu items are on the left, including my score menu, course information query menu, employment menu, payment menu, reward and punishment menu, and password modification menu. These menu items have different functions but are all related to course information management. Click each menu, and the system will enter the corresponding next sub-interface. For example, clicking on my grades menu will bring you into the interface for viewing personal grades, Clicking the course information query menu to query the specific course information, Clicking on my employment menu to view personal employment information, etc. These sub-interfaces provide straightforward operation tips and functional options so users can efficiently perform various operations. On the right side of the course information interface, the specific course name and the corresponding information, such as class time, class place, and class, are displayed. This information is clear at a glance so that users can quickly understand the relevant information of the course. At the same time, the interface design is concise and clear, and the operation is simple and easy, which makes course information management easy and convenient.

To verify the load balance of the server cluster in the present work, the experiment uses the requesting user to log in to the system for operation. This request not only has a static page but also has a database operation. It belongs to a dynamic request. The system feedback data analyzes the average response time of the request and the number of successful requests. The experimental results of comparing the system in this paper, using the DVFS computing system and cloud system task scheduling system, are shown in Fig. 11 and Fig. 12.

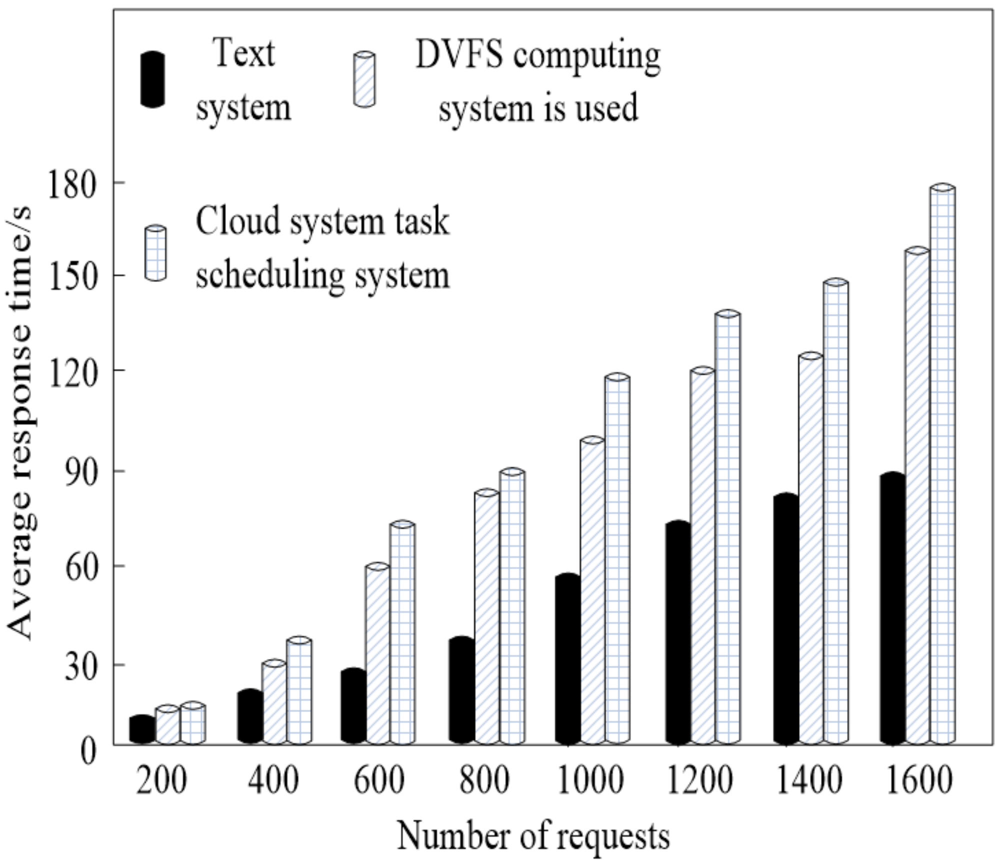


Fig. 11. Average response time for the three system requests.

The experimental results in Fig. 10 show the average response time performance of the three systems when processing different requests. When the number of requests is 200, the average response time of the three systems is within the acceptable range, and the difference between them is not apparent. However, with the increase in the number of requests, the average response time of the system in this paper gradually increases but remains in a very short range. When the number of requests reaches 1600, the average response time of the system in this paper only takes 90 seconds, which has significant advantages over the other two systems. With the DVFS computing system, the average response time has reached 120 seconds when the number of requests is 1200, and the average response time of the cloud system task scheduling system is 150 seconds when the number of requests is 1400. To sum up, compared with the other two systems, the average response time of the system in this paper is shorter when processing a large number of requests, which shows that the server cluster in this system is more balanced and stable and has higher efficiency and better performance when processing large-scale parallel computing tasks. This will provide higher vocational colleges with faster, more efficient, and more reliable student information management services.

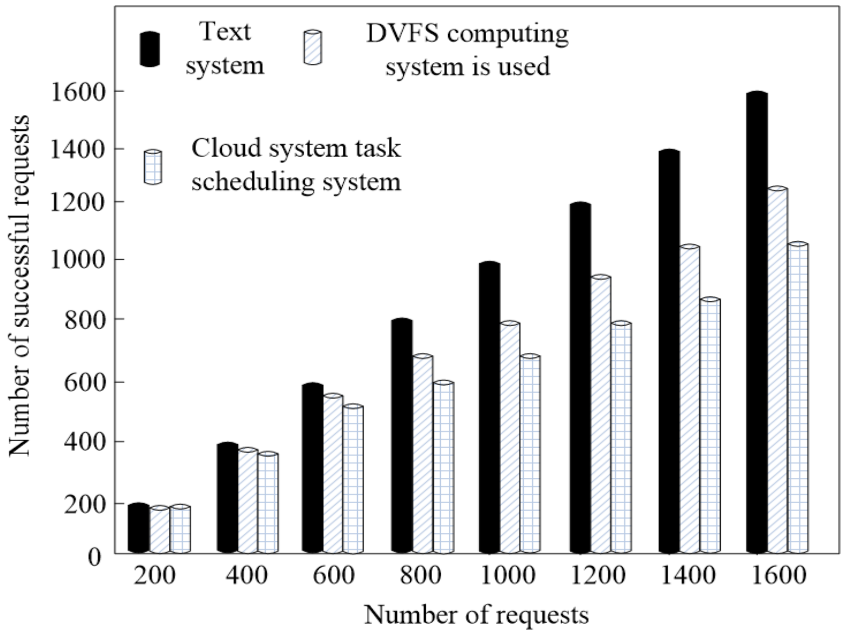
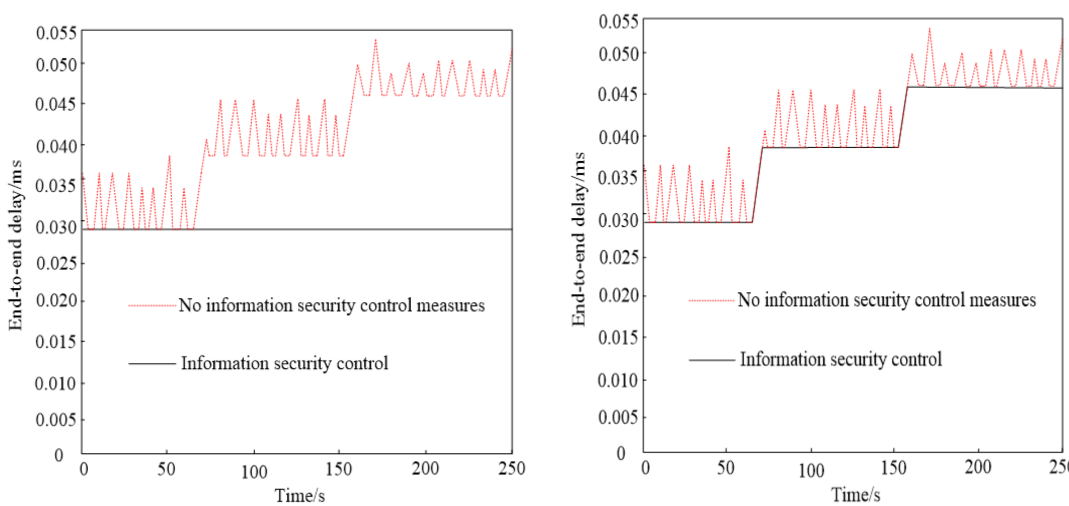


Fig. 12. Number of successful requests for the three systems.

Analyzing Fig. 12 shows that when the system in this paper processes many requests, the number of successful requests is not affected, and all requests are successful. This indicates that the system in this paper has excellent stability and reliability. In contrast, when the DVFS computing system reaches 1000 requests, the number of successful requests is only 800, which shows obvious instability. When the number of requests of the cloud system task scheduling system reaches 1200, the number of successful requests is only 800, and its performance is worse. This data shows that the system can maintain the stability and balance of server cluster load and avoid large fluctuations when dealing with large-scale parallel computing tasks. Hence, this system can fulfill the requirements of the student information management system and guarantee its consistent functionality. In conclusion, the system discussed in this document offers notable benefits in managing many simultaneous requests, catering to the student information management system's demands, and upholding the server cluster's stability.

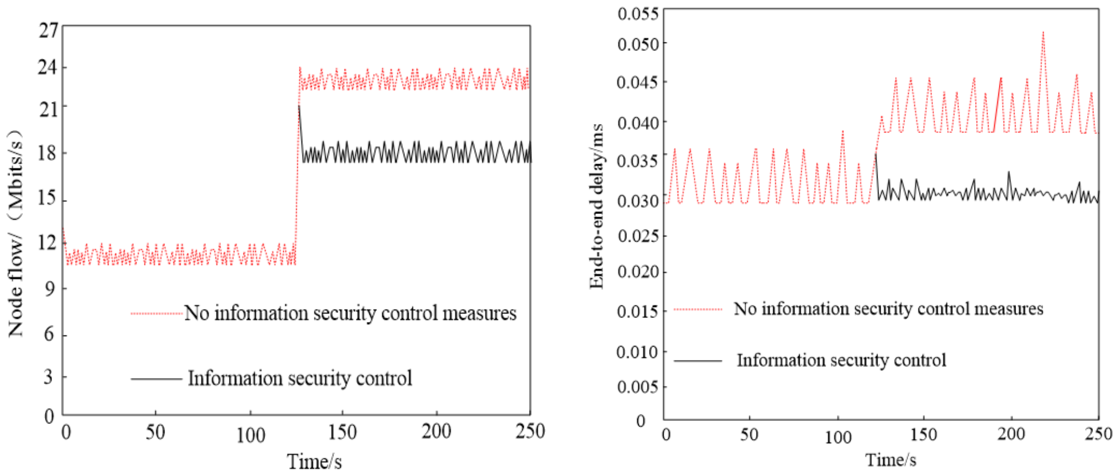
To verify the security control of the integrated virtual gateway applied in this system on the information management of college students in higher vocational colleges, the following four situations are used for verification: non-critical access request burst operation, critical access request burst operation, central exchange node traffic, and end-to-end delay of access request. The first two are used to verify the information security control of the integrated virtual gateway to normal network communication. The last two are used to verify the abnormal operation of the integrated virtual gateway to the network, that is, the network communication is invaded or maliciously controlled from the outside, which leads to the abnormal growth of information and network storm. The results are shown in Fig. 13 and Fig. 14.



(a) Non-critical access requests burst (b) Non-critical access requests burst

Fig. 13. Information security control of normal network communication.

It can be seen from Fig. 13 that under the normal operation of the network communication without abnormal intrusion, after the information security control by the system in this paper, the critical access request delay is stable at around 0.029ms, without noticeable delay jitter. Even in the case of two necessary access request bursts, the delay finally increases to 0.046ms and remains stable. The delay jitter is evident in the absence of information security control measures. This shows that the introduction of the integrated virtual gateway in this system is efficient for the information security control of normal network communication, which proves that the information security control of this system can effectively ensure the normal operation of the network, reduce delay jitter, and improve the stability and reliability of network communication.



(a)Traffic of central switch nodes (b)End-to-end latency of access requests

Fig. 14. Information security control of abnormal network communication.

As shown in Fig. 14(a), the node traffic of the central switch is stable when the student information management system is running normally. In the period of 0-130 seconds, the information security control measures applied to the system in this paper did not significantly impact the switch traffic distribution, and the traffic was stable in the range of 11.5-12Mbits/s. However, when the system is between 130-250 seconds when it is invaded externally, without taking information security control measures, the traffic of the central switch increases sharply, reaching 22.5-24 Mbits/s. This will lead to network congestion, performance degradation, and even paralysis. Under the condition of taking information security control measures, the traffic of the central switch is significantly reduced. This shows that the information security control of this system effectively alleviates network congestion and avoids the risk of network storms and network paralysis. Similarly, in the end-to-end delay result of the access request shown in Fig. 10 (b), the information security control of this system can maintain the relatively stable access request delay under abnormal conditions. The demonstration above illustrates the system's capability to ensure the dependable execution of the student information management system and maintain the system's smooth functioning even in exceptional circumstances. This serves as evidence that the information security control measures implemented in this system have the potential to significantly enhance the network's stability and reliability, thereby ensuring the uninterrupted operation of the student information management system.

B. *Discussion*

In conclusion, the cloud computing-based information management system for college students designed in this paper shows excellent performance in processing massively parallel computing tasks, and the average response time increases slowly, which ensures efficient processing. At the same time, the system keeps the success of requests as high as 100%, stably and reliably meet the needs of students' information management. By optimizing the load balancing of the server cluster, the stability of the system in this paper far exceeds the literature [4] system and the literature [5] system, providing a solid guarantee for the stable operation of the school. In addition, the introduction of integrated virtual gateway realizes the accurate monitoring and efficient management of network communication, which greatly improves the stability and reliability of network communication. In conventional communication conditions, the system significantly reduces the delay jitter of key access requests to ensure the smooth network communication. In abnormal cases, information security control measures can effectively alleviate network congestion, avoid the risk of network storm and paralysis, and ensure the reliable implementation of the student information management system.

IV. CONCLUSION

This paper uses cloud computing technology to present a novel information management system for higher vocational college students. The experimental findings demonstrate that the student information management system proposed in this study possesses a user-friendly interface design, facilitating convenient and efficient operations. Moreover, it offers comprehensive and diverse management functions, ensuring efficiency, flexibility, and security. Consequently, this system effectively caters to the information management needs of higher vocational colleges. By implementing this system, higher vocational colleges' work efficiency and management standards can be enhanced, leading to improved student services. Additionally, the system significantly saves human and material resources, improving the institution's overall operational efficiency. In conclusion, the successful implementation of this system not only reflects the great potential of cloud computing technology in the field of student information management, but also provides a useful reference for the design and development of other similar systems. In the future, with the continuous progress of technology and the deepening of its application, it is believed that the student information management system based on cloud computing will be more perfect, efficient and intelligent, and bring more convenience and value to the education management of higher vocational colleges.

Competing of Interests

The authors declare no competing of interests.

Authorship Contribution Statement

Mo Bin: Writing-Original draft preparation, Conceptualization, Supervision, Project administration.

Data Availability

On Request

Declarations

Not applicable

References

[1] M. H. Abidi, H. Alkhalefah, U. Umer, and M. K. Mohammed, "Blockchain-based secure information sharing for supply chain management: optimization assisted data sanitization process," International journal of intelligent systems, vol. 36, no. 1, pp. 260–290, 2021.

[2] S. Sengupta, J. Garcia, and X. Masip-Bruin, "Essentiality of managing the resource information in the coordinated fog-to-cloud paradigm," International Journal of Communication Systems, vol. 33, no. 10, p. e4286, 2020.

[3] V. Hakkoymaz and C. Bakir, "Information Flow Control with Decentralized Labeling Model in Information Security," Journal of Web Engineering, Special Issue: Advanced in Al and Nature-inspired Approaches for Web Data Security, vol. 19, no. 7–8, pp. 1–23, 2020.

[4] D. C. Li, B.-H. Chen, C.-W. Tseng, and L.-D. Chou, "A novel genetic service function deployment management platform for edge computing," Mobile Information Systems, vol. 2020, pp. 1–22, 2020.

[5] K. Ameri, M. Hempel, H. Sharif, J. Lopez Jr, and K. Perumalla, "Design of a novel information system for semi-automated management of cybersecurity in industrial control systems," ACM Trans Manag Inf Syst, vol. 14, no. 1, pp. 1–35, 2023.

[6] D. Bi, S. Kadry, and P. M. Kumar, "Internet of things assisted public security management platform for urban transportation using hybridised cryptographic-integrated steganography," IET Intelligent Transport Systems, vol. 14, no. 11, pp. 1497–1506, 2020.

[7] W. Chen, R. Samuel, and S. Krishnamoorthy, "Computer Vision for Dynamic Student Data Management in Higher Education Platform.," Journal of Multiple-Valued Logic & Soft Computing, vol. 36, 2021.

[8] D. G. Rosero, N. L. Díaz, and C. L. Trujillo, "Cloud and machine learning experiments applied to the energy management in a microgrid cluster," Appl Energy, vol. 304, p. 117770, 2021.

[9] R. Dhaya, R. Kanthavel, and K. Venusamy, "AI Based Learning Model Management Framework for Private Cloud Computing," Journal of Internet Technology, vol. 23, no. 7, pp. 1633–1642, 2022.

[10] J. K. Samriya, S. Chandra Patel, M. Khurana, P. K. Tiwari, and O. Cheikhrouhou, "Intelligent SLA-aware VM allocation and energy minimization approach with EPO algorithm for cloud computing environment," Math Probl Eng, vol. 2021, pp. 1–13, 2021.

[11] J. Kittur, J. Bekki, and S. Brunhaver, "Development of a student engagement score for online undergraduate engineering courses using learning management system interaction data," Computer Applications in Engineering Education, vol. 30, no. 3, pp. 661–677, 2022.

[12] D. Demırkol, C. Seneler, T. Daım, and A. Shaygan, "Measuring perceived usability of university students towards a student information system (SIS): A Turkish university case," Technol Soc, vol. 62, p. 101281, 2020.

[13] N. El Ioini, H. R. Barzegar, and C. Pahl, "Trust management for service migration in Multi-access Edge Computing environments," Comput Commun, vol. 194, pp. 167–179, 2022.

[14] M. Hassannezhad, B. Farahany, and F. Barzegar, "Virtual Net Propagator: A cloud-based computational tool for systemic decision propagation analysis," Expert Syst Appl, vol. 191, p. 116338, 2022.

[15] B. Vuppala and P. Swarnalatha, "Software defined network using enhanced workflow scheduling in surveillance," Comput Commun, vol. 151, pp. 196–201, 2020.

[16] A. Javadpour et al., "An energy-optimized embedded load balancing using DVFS computing in cloud data centers," Comput Commun, vol. 197, pp. 255–266, 2023.

[17] P. Aruna Kumari and I. Santi Prabha, "Hierarchical and hybrid cell load balancing in 5G heterogeneous mobile networks," International Journal of Communication Systems, vol. 35, no. 1, p. e5017, 2022.

[18] F. S. Nininahazwe, J. Shen, and M. E. Taylor, "An Augmented Load-Balancing Algorithm for Task Scheduling in Cloud-Based Systems," Journal of Internet Technology, vol. 22, no. 7, pp. 1457–1471, 2021.

[19] A. Moori, B. Barekatain, and M. Akbari, "LATOC: an enhanced load balancing algorithm based on hybrid AHP-TOPSIS and OPSO algorithms in cloud computing," J Supercomput, vol. 78, no. 4, pp. 4882–4910, 2022.

[20] R. Rabaninejad, M. Rajabzadeh Asaar, M. Ahmadian Attari, and M. R. Aref, "An identity-based online/offline secure cloud storage auditing scheme," Cluster Comput, vol. 23, pp. 1455–1468, 2020.

[21] Y. M. Gajmal and R. Udayakumar, "Blockchain-based access control and data sharing mechanism in cloud decentralized storage system," Journal of web engineering, vol. 20, no. 5, pp. 1359–1388, 2021.

[22] P. Sharma, R. Jindal, and M. D. Borah, "Blockchain technology for cloud storage: A systematic literature review," ACM Computing Surveys (CSUR), vol. 53, no. 4, pp. 1–32, 2020.

[23] C. Yu and Z. Zhao, "Information storage optimization of electronic archive system based on information fusion," Comput. Integr. Manuf. Syst, vol. 38, no. 6, pp. 455–459, 2021.

[24] T. An, "Data mining analysis method of consumer behaviour characteristics based on social media big data," International Journal of Web Based Communities, vol. 18, no. 3–4, pp. 224–237, 2022.

[25] M. Gupta and P. Kumar, "Recommendation generation using personalized weight of meta-paths in heterogeneous information networks," Eur J Oper Res, vol. 284, no. 2, pp. 660–674, 2020.

[26] A. Zyane, M. N. Bahiri, and A. Ghammaz, "IoTScal-H: hybrid monitoring solution based on cloud computing for autonomic middleware-level scalability management within IoT systems and different SLA traffic requirements," International Journal of Communication Systems, vol. 33, no. 14, p. e4495, 2020.