

Research Article

Edge Computing in Mobile Information System for Digital Construction of College English Teaching

Hong Zhu 

College of Foreign Languages and International Education, Dalian Ocean University, Dalian, 116023 Liaoning, China

Correspondence should be addressed to Hong Zhu; zhuma@dlou.edu.cn

Received 18 October 2021; Revised 16 November 2021; Accepted 18 November 2021; Published 21 December 2021

Academic Editor: Deepak Kumar Jain

Copyright © 2021 Hong Zhu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As an education mode and teaching method of modern education, digital teaching in colleges and universities plays a role of bridge and hub in the development of modern education. Digital teaching has become one of the important driving forces for the development of digital and informatization of education in colleges and universities. This paper combines edge computing and digital technology to study the mobile information system used in the digital teaching of college English, namely, the digital teaching platform. In the technical part, this article clarifies the concept of mobile information system and introduces edge computing, cloud computing, digital system structure, and other technologies in detail; the system design and implementation part introduces the system architecture, function design, and how to realize the system function. In the system test section, black box and white box tests were performed on the system. The test results show that all functions of the system can be used normally and have certain feasibility. The mobile information system reduces the average content transmission delay (ADL) by approximately 14%-22% and 21%-31%, respectively, which greatly improves the efficiency of digital teaching of English in colleges and universities.

1. Introduction

1.1. Research Background. Nowadays, digital construction is rising with the rapid development of Internet technology and information systems. Digital construction is constantly advancing in all fields of society. Teaching is an important field of digital construction. The opportunities and challenges brought by digital construction have increasingly impacted traditional teaching and management models. The problems and tasks faced by colleges and universities in teaching and educational administration management are constantly increasing. Traditional low-efficiency teaching management methods have development limitations. Therefore, the digitalization and information management of teaching have become the trend of future development. At present, society is gradually moving towards the era of “big data,” teaching in colleges and universities is gradually digitizing, and technologies such as cloud computing and edge computing have emerged. The emergence of cloud computing and edge computing has increased the ability to process big data [1].

Cloud computing and edge computing technologies have the advantages of low operating costs, dynamic scalability, and simplified operation and maintenance. Therefore, they have been vigorously promoted by various industries in the society, and their related industries have also developed rapidly.

1.2. Research Significance. The mobile information system is of great significance to the digital construction of college English teaching. If we can combine the edge computing of mobile information system and the digital teaching of college English, then university education can be made more scientific and digital. Its research and design and realization have practical value and promotion significance. Its significance is mainly reflected in the realization of information sharing on the one hand, and the main purpose of digital teaching resource information is to achieve resource sharing; the mobile information system can realize the centralized management of effective information modules and integrate data into the mobile information platform so that each module realizes the sharing of teaching resources; on the other

hand, it improves the efficiency of teaching and office, and the digital teaching information system integrates various functions. Module integration, construction of an integrated information platform, simple operation, and convenient use can greatly improve the efficiency of digital teaching management.

1.3. Related Work. Nowadays, the digitalization of college education has begun to incorporate network cloud technology [2] and edge computing technology [3]. First, the basic concepts and theories are introduced. Secondly, it introduces the construction process of the college English teaching mobile platform and the effect realization process of the mobile terminal and uses the fuzzy neural network for intelligent calculation. Finally, investigations and tests were conducted. The results show that the teaching digital mobile platform constructed in this paper can provide good teaching effects for English education [4]. Gong Rui, Shi Sheng, and Sun Yi insisted that the core of the construction of a teaching management information system in colleges and universities is an innovative concept. The talent training plan of colleges and universities is the starting point of the teaching management process and the basis of teaching management evaluation and decision analysis. The digital teaching information system of colleges and universities should take the talent training plan as the core, design the digital teaching information system architecture, and organically integrate the digital platform with the teaching platform and the information system architecture for development and design. Digital teaching mobile information system: perfect teaching management process and results, perfect management and service, perfect teaching management, and perfect information collection. Realize the integration of teaching management process and results, the integration of management and service, the refinement of teaching management, and the integration of information collection. The theory and thought process of designing a digital teaching system with education as the core can be used as an important reference for the construction of digital teaching in colleges and universities [5]. As an important means for colleges and universities to improve their English teaching ability, digital English teaching is conducive to improve the level of English teaching. For this reason, Liu conducted a research on the evaluation of college English classroom teaching level. Establish an evaluation index system by analyzing the selection principles of the evaluation index of English classroom teaching level. Then, classify this kind of evaluation, and calculate the fuzzy membership degree of each index of the evaluation object at different levels. Then, use the analytic hierarchy process to obtain the corresponding evaluation index weights, and establish a multi-index measurement model of college English classroom teaching level based on the fuzzy system theory. The results of the research show that the proposed model is feasible [6]. Traditionally, parasitology courses are mostly taught face-to-face on campus, but now, digital technology provides opportunities for teaching and learning. Here, Karakoyun presents his views on how to use new technologies through student-centered teaching methods. First, a brief

introduction to the latest trends in higher education; then, a brief introduction to how digital technologies (for example, Mass Open Online Courses (MOOC), flipped classrooms (FC), games, quizzes, dedicated Facebook, and digital badges) promote digital learning teaching and learning of parasitology in the environment. He believes that some of these digital technologies may contribute to ability-based, self-regulating, and learner-centered teaching and learning in online or mixed teaching environments [7]. In recent years, the Internet of Things (IOT) has gradually changed the world as a new disruptive technology. In order to reduce user delay and improve user experience, while reducing network load to a certain extent, edge computing emerged as an application of the Internet of Things. Aiming at the new architecture after dating edge computing, Zhang starts from task migration in multiuser scenarios and the expansion of edge server resource management, focusing on task offloading in edge computing [8]. But they are not going to build a mobile information platform for digital education.

1.4. Innovation. Mobile information system is an important support and tool for the construction of digital teaching in colleges and universities, and this article combines cloud computing, edge computing, and other technologies for the construction of mobile information systems and the construction of digital teaching in college English. This paper designs an English teaching platform by analyzing the architecture and model of mobile edge computing and then uses virtual machine allocation algorithms, virtual machine dynamic scheduling algorithms, and target detection and recognition algorithms under edge computing to improve the specific functions of the English teaching platform. In the technical part, traditional and modern mobile information systems and mobile edge computing technology (mobile edge computing (MEC)) are introduced, and today's digital teaching methods and two system architectures are also introduced. In the part of system design and implementation, the teaching system has been designed with architecture design, function design, database design, etc., using programming methods to achieve system-related functions and testing and analysis of the system. The innovation of this paper is to combine mobile information systems with cloud computing and edge computing to apply it to the research of digitization of English teaching in colleges and universities and to promote the realization of digitized teaching and digital campus in colleges and universities.

2. Edge Computing in the Mobile Information System Is the Relevant Technology for the Digital Construction of College English Teaching

2.1. Mobile Information System. The mobile information system used for digital teaching in colleges and universities refers to the information service system that uses digital technology to collect, store, process, transmit, and share teaching information resources in colleges and universities, thereby realizing digital teaching and learning. The mobile

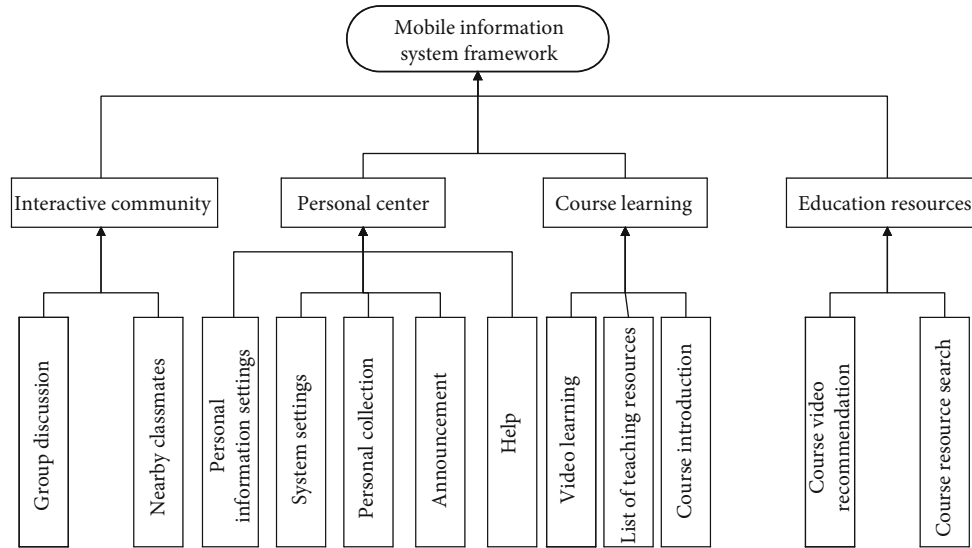


FIGURE 1: The framework of the mobile teaching information system.

information system uses advanced information technologies such as computers, databases, and communication networks to collect, store, process, transmit, and share teaching information resources in colleges and universities. The ultimate goal is to realize the digitization and informatization of teaching management; Figure 1 shows the framework of the mobile teaching information system.

IMIS looks at information management issues from a holistic perspective, using the hub of teaching management information to manage, coordinate, and control the various modules of the digital teaching information system and professionalize and digitize the problems in teaching management. Research, discuss, supplement, and improve the digital teaching theory system from different perspectives. The theoretical basis of the system—modern management theory, education management theory, statistical education theory, system theory, information theory, computer science, and theoretical development and perfection—is the IMIS theory of the construction and definition of the concept provide strong support.

Teaching management includes teaching plan and lesson preparation, timetable management, examination management, teaching material management, and teaching resource management. It is characterized by large amount of information, complex processing, more daily dynamic information, and high requirements for the timeliness and sharing of information resource transmission. The efficiency and effectiveness of teaching management are affected by it, and teaching management information is effective. As the traditional teaching management is mostly manual management or independently operated management system, with the development of digital technology and information system, the teaching of my country's colleges and universities faces many new problems and challenges, such as the gradual expansion of teaching scale, the widespread application of information technology, the credit system management operation, and the emergence of online teaching mode.

The scope of teaching management continues to expand, the amount of information that needs to be processed increases sharply, and the sources and forms of information are increasingly diversified, which puts forward higher requirements for the rigor, accuracy, and timeliness of teaching management.

2.2. Mobile Edge Computing

2.2.1. Architecture and Model of Mobile Edge Computing.

From the macro level, MEC can be divided into 3 levels according to the different functional entities of MEC: mobile edge system layer, mobile edge host layer, and network layer [9]. From top to bottom, the first layer is the mobile edge system layer, which is responsible for controlling the global MEC system and is the management entity of the MEC system level. The second level is the mobile edge host layer, which includes mobile edge applications and mobile platforms, virtualized infrastructure, mobile edge management systems, and mobile edge host layer management systems at the mobile edge layer. The last layer is the network layer. The network layer mainly includes related external entities such as 3GPP cellular network, local area network, and external network. This layer mainly represents the MEC operating system and the access status of the local area network, mobile network, or external network. Among them, the mobile edge host can access the network on the basis of the network level, and the network level must be at the mobile edge host level to play its low latency and high quality characteristics. The mobile edge system is responsible for the overall management of the mobile edge host level and network level. The three are interdependent and closely related, and each plays an important role.

The three layers are interdependent and closely related, and each plays an important role. The basic architecture of MEC is shown in Figure 2.

The basic model of MEC is shown in Figure 3.

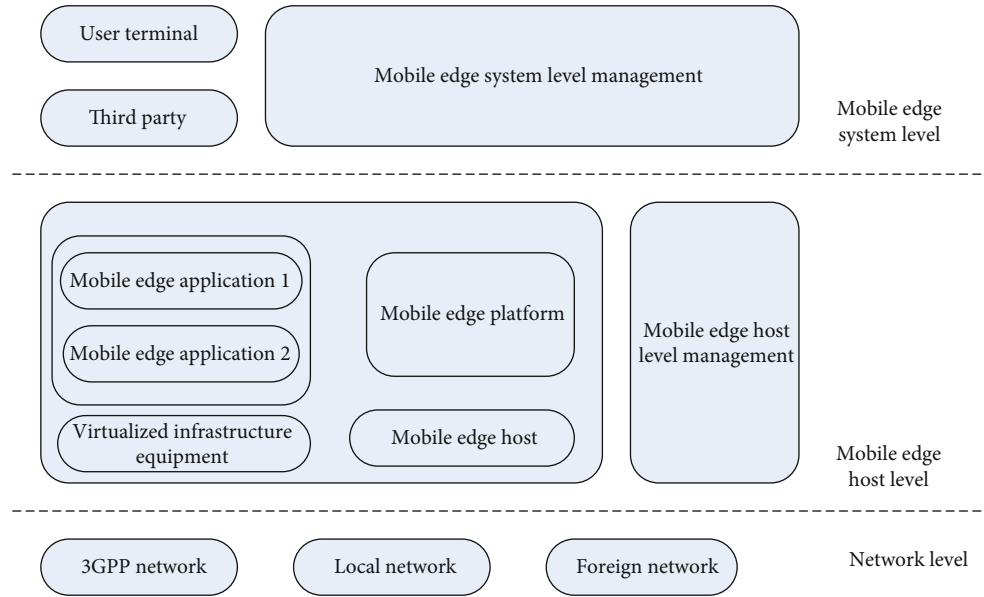


FIGURE 2: Basic architecture of MEC.

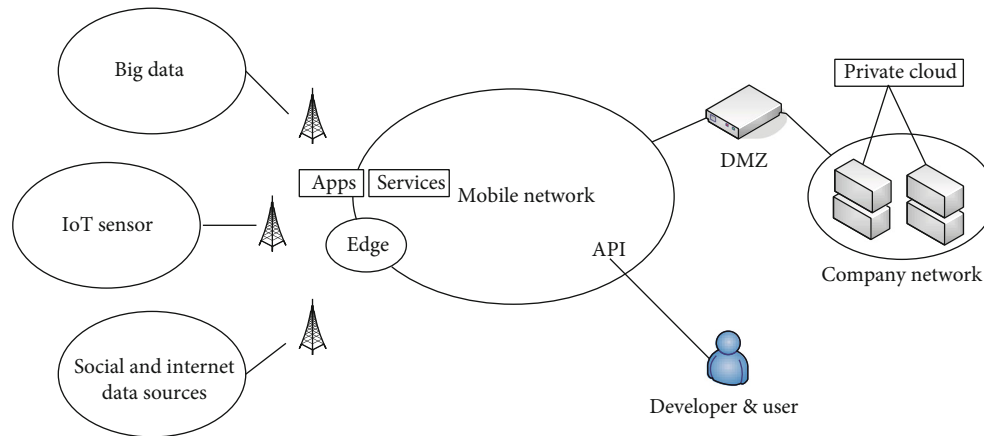


FIGURE 3: MEC model.

2.2.2. *Key Technologies of Mobile Edge Computing.* MEC key technologies are mainly divided into the following three points:

- (1) **Computing unloading technology:** based on this technology, the MEC system can efficiently schedule tasks, greatly reduce the download terminal delay, and achieve real-time business processing. The core of real-time business processing in the MEC system is computing offloading technology. After downloading the terminal, the programmed MEC server executes and allocates corresponding computing resources. How to schedule tasks to be executed on suitable servers and how to allocate computer resources to terminals and servers reasonably are the key to computing offloading technology. The Lyapunov function [10] can be used to solve such optimality problems and the upper algorithm, and

ϵ -greedy algorithm [11] can be used to solve the unloading optimization problem

- (2) **Wireless data caching technology:** the MEC system supports this technology, so the MEC system can support local cache applications to reduce the delay of mobile terminal delivery, reduce service delays, and improve user experience. The basic principle of this technology is to assume that the hotspot data is cached on the MEC edge server. If the user requests the hotspot data again, the user can obtain the required data within a single hop range, thereby reducing the delay [12]
- (3) **Local offload technology based on SDN network:** the local offloading technology is used to solve applications such as video, monitoring, and live broadcasting locally and realize the offloading of services

locally, thereby reducing delay. Use SDN's local offloading technology to separate the control layer and data layer of the SDN network and achieve intelligent offloading in the offloading gateway to achieve local offloading. The whole process is divided into three steps: in the first step, the SDN controller obtains the corresponding diversion strategy from the server; in the second step, in compliance with the diversion strategy and network flow information, the SDN controller formulates the corresponding diversion rule table and then sends it to the diversion gateway; in the third step, the shunt gateway performs final shunt according to the shunt forwarding rule table [13]

2.3. Edge Computing-Related Algorithms

2.3.1. Virtual Machine Allocation Algorithm. The goal is to put multiple virtual machines (objects) into multiple virtual nodes (boxes) and minimize the number of virtual nodes occupied and the load variance value [14]. The allocation problem of multitarget virtual machines can be described in the formula as follows:

$$\begin{aligned} f_{PM} &= \min \sum_i C_i, \\ f_{LN} &= \min \frac{\sum_c D\xi_j}{C}. \end{aligned} \quad (1)$$

In the above two formulas, f_{PM} represents the number of occupied virtual nodes and f_{LN} represents the variance of the load balance of the server cluster.

i represents the number of virtual nodes, \bar{P}_i represents the average value of the i -th dimension performance characteristics of all virtual nodes, and the performance characteristic is a normalized value, which is equal to the remaining allocated amount of the i -th dimension resources in the virtual node divided by the total i -dimensional resource amount. P_{ij} is the performance characteristic value of the i -th dimension of virtual node j [15], and its expression is

$$D\xi_i = \frac{\sum_n (P_{ij} - \bar{P}_i)}{n}. \quad (2)$$

VM_m^{CPU} , VM_m^{nen} , and VM_m^{IO} are the CPU, memory, and data transmission resources of the virtual machine m ; P_n^{CPU} , P_n^{nen} , and P_n^{IO} are the CPU, memory, and data transmission resources of virtual node j [16], and the formula is as follows:

$$\begin{aligned} \sum_m VM_m^{CPU} * X_{m \rightarrow n} &\leq P_n^{CPU}, \\ \sum_m VM_m^{nen} * X_n &\leq P_n^{nen}, \\ \sum_m VM_m^{IO} * X_n &\leq P_n^{IO}. \end{aligned} \quad (3)$$

The application system P is deployed on cn virtual machines. In the SLA (service level agreement) of each application system P , we will meet the performance indicators of the application system P to meet the cn virtual machine VMs and the dynamic resource scheduling of cloud computing. The specific expression of algorithm [17] is

$$\sum_{cn} VM_{ip} \geq SLA_p. \quad (4)$$

The two goals of load balancing are as follows: one is to ensure the high performance of the application system; the other is the low utilization rate of virtual nodes [18]. Through two fitness functions, the formula for evaluating individuals is as follows:

$$Fit = \left\{ \min \sum_n C_n, \min \frac{\sum_d D\xi_m}{d} \right\}. \quad (5)$$

2.3.2. Virtual Machine Dynamic Scheduling Algorithm. A performance-based approach is adopted to measure the overall load value of the virtual node, which is compared with the initially set maximum load critical value D_{max} and minimum load critical value D_{min} to determine [19]; the total load value of the virtual node is

$$CP_i = \frac{\sum_{j=0}^i Q_{ij}}{m}. \quad (6)$$

The dynamic migration of virtual machines mainly includes the running state and data resources, and the migration of memory occupies a large part of the dynamic overhead. Therefore, the overall impact of dynamic overhead should be considered when determining the virtual machine [20]. Define the variable dynamic loss ratio as

$$cn = \frac{C_i + dt_i}{n_i}. \quad (7)$$

Assuming that the rate of various operations entering the edge service is [21], the probability of each operation appearing is

$$Z_\omega = \frac{\lambda_\omega}{\sum_\omega \lambda_\omega}. \quad (8)$$

The time within the coverage of the hierarchical mobile edge service [22] can be expressed as

$$T_\omega = \frac{2R}{v_\omega}. \quad (9)$$

2.3.3. Performance Index. This article compares three performance indicators, namely, the cache hit rate (hit rate (HR)), the average content delivery latency (average delivery latency (ADL)), and the average content transmission cost (average transmission cost (ATC)). Thus, the cache hit rate (HR) [23]

can be expressed as

$$\text{HR}^{\text{def}} = \frac{R-M}{R} = 1 - \frac{M}{R}. \quad (10)$$

The average content transmission cost (ATC) [24] index is expressed as

$$\text{ADL} = \sum_{m=1}^L M \sum_{i=1}^L P_{m,i} * (y_{m,i}^1 + y_{m,i}^2). \quad (11)$$

2.3.4. Target Detection and Recognition Algorithm under Edge Computing. Faster R-CNN target detection algorithm: the algorithm obtains the midpoint coordinates of the anchor point box and the length and width of the anchor point box and performs linear regression with the four data of the anchor point box. The formula is as follows:

$$\begin{aligned} f_x &= \frac{x - x_\alpha}{\omega_\alpha}, \\ f_y &= \frac{y - y_\alpha}{h_\alpha}, \\ f_\omega &= \lg \left(\frac{\omega}{\omega_\alpha} \right), \\ f_h &= \lg \left(\frac{h}{h_\alpha} \right), \\ f_x^* &= \frac{x^* - x_\alpha}{\omega_\alpha}, \\ f_y^* &= \frac{y^* - y_\alpha}{\omega_\alpha}. \end{aligned} \quad (12)$$

The performance and effect of Faster R-CNN target detection algorithm can basically be used in the actual environment, and its detection speed can reach 16 fps.

2.4. Digital Teaching. The design and implementation of college English digital teaching platform should pay attention to the following aspects.

- (1) Taking the creation of an adaptive language learning environment as the starting point, the key to building a digital English teaching platform is to build an effective online and offline learning environment that seamlessly connects the meaning of the advanced design classroom inside and outside. It is necessary to improve the teaching environment and change the teaching structure and teaching methods on the basis of digital technology. With digital and information as the center, technology can be applied anytime and anywhere according to the needs of teaching
- (2) Integration of multiple technologies tools and functions: on the one hand, the design of the English digital teaching platform should develop corresponding functional modules according to the needs of stu-

dents' language learning, support the organization of multiple learning activities and the development of personalized learning, and focus on coordination and integration internal function. On the other hand, to effectively utilize platform functions, it is necessary to strengthen the organic combination of platform tools and social tools

- (3) Strengthen the dynamic management of the construction of the English digital teaching platform and the integration of learning resources, realize the sharing of internal and external resources on the platform, and strengthen the effective integration of various resources within the platform
- (4) Platform construction should not only focus on one-way skill training and process support but also strengthen the design of activity support function modules for collaborative training of various skills such as listening, speaking, reading, and writing

2.5. System Structure

2.5.1. B/S Structure. B/S stands for Browser/Server, which is a system architecture model based on browsers and servers. Therefore, the B/S structure is generally widely used in the Internet system environment, especially for the construction of dynamic network systems; the advantages of the multi-layer architecture based on the B/S structure can be fully reflected. The B/S structure is based on the transformation of the C/S structure, which splits the server in the C/S structure into a database server and a Web server, and the three-tier structure of the B/S comes from this. Therefore, the upper B/S structure is composed of the presentation layer, the logic layer, and the data layer [25, 26], and its structure is shown in Figure 4.

2.5.2. C/S Structure. C/S stands for Client/Server, which is a system architecture model based on client and server. The C/S architecture is generally a two-tier structure. The client is responsible for interacting with the user, and the server is responsible for managing data. The client connects to the server through the network, accepts user requests, and then submits the request to the server to operate the database. The server accepts the client's request and submits the data to the client, and the client calculates the data and presents the result to the user [27]. Its structure is shown in Figure 5.

3. Design of Mobile Information System for Digital Teaching in Colleges and Universities

3.1. Demand Analysis. The users of this system are teachers and students. When investigating the needs, I found that the teaching system cannot be accessed in places that are not connected to the campus network, and it is impossible to query information and process business in time. Teachers and students urgently need an application that can accurately query information and process business on mobile smart terminals anytime and anywhere. Summarize the

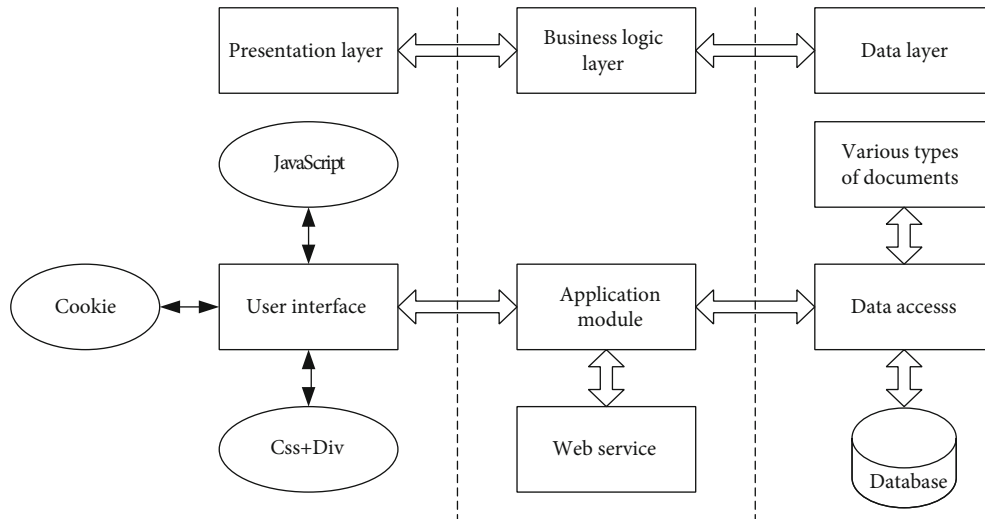


FIGURE 4: B/S structure diagram.

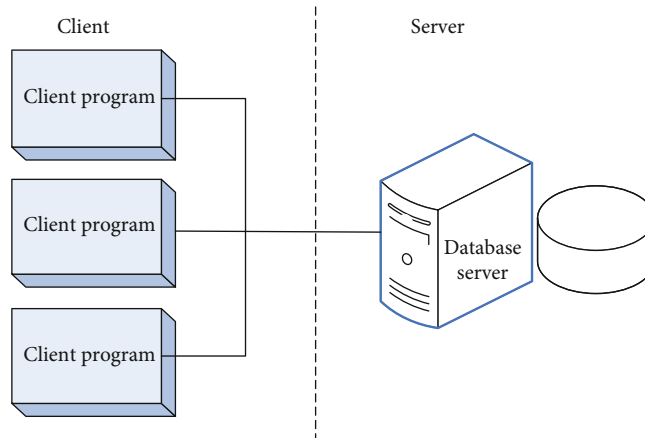


FIGURE 5: C/S structure diagram.

needs of the research, solve the specific needs of the system, and get the system use case diagram as shown in Figure 6.

3.2. *Overall Architecture.* The data sharing and exchange platform integrates the original standalone application system, teaching platform, resource platform, management platform, and campus service platform database so that it can be integrated in a mixed environment of different systems. The platform integrates the data supported by the system database into the platform through ETL tools; after conversion, debugging, and standardization, it is finally imported into the shared database for real-time or timing synchronization [28, 29]. The overall architecture of the data sharing and exchange platform is shown in Figure 7.

4. Realization of Digital Teaching Mobile Information System in Colleges and Universities

4.1. *System Structure.* The system uses mobile technology to propose a mobile information system solution using wireless

networks and mobile smart terminals. The mobile phone installs the system application, the user sends a request to the server through the wireless network on the mobile information platform, and the server forwards the request to the mobile information platform server. The platform server interacts with the school data exchange platform and then returns the data requested by the client to the server, and finally, the server forwards the data to the client. The architecture of the system is shown in Figure 8.

4.2. *Technical Selection.* This system adopts C/S architecture. The system server is implemented in Java language, using Servlet to receive all client requests forwarded by the application server, then call the corresponding Java-Bean according to the requested content and forward the result to the application server, and finally send it from the server to the user. The local database of the system uses MySQL. Use Hibernate technology to map data. The web server uses Tomcat. Transfer data between the application server and the teaching platform server in the specified XML format.

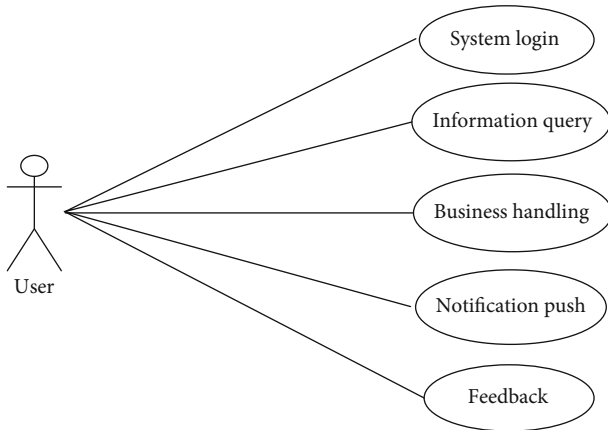


FIGURE 6: System use case diagram.

4.3. Feature Design. Through the demand analysis of the system, the system is divided into five functional modules: identity authentication, information inquiry, business handling, notification sending, and opinion feedback. The digital teaching mobile information platform mainly realizes the functions of platform login, timetable query, and teaching resource query. In addition, the platform also implements notification sending and opinion feedback functions. The system only realizes some query functions such as grade query, class schedule query, and borrowing query. The four functional modules of identity authentication, information inquiry notification sending, and opinion feedback will be analyzed and designed in detail below according to the characteristics of the digital teaching platform.

4.4. Database Design. System data is mainly obtained from the data sharing and exchange platform, and the local database only stores user information, menus, messages and notifications. Based on the functions required by the system, the teacher table, student information table, class table, menu table, and notification table are designed, focusing on the student information table and menu table.

4.4.1. Student Information Form. The student information table is mainly used to store student information, menu status, and binding status. Every time a user connects to the platform, he must determine the user's binding status. Only users who have bound their student ID and WeChat ID can use some functions. The E-R diagram of student user information is shown in Figure 9.

Through the E-R diagram, design the main fields of the student table. The bind field is used to store the user's binding status; the default is 0, and if it is bound, it is 1. The status field is used to store the user's menu status; the default is 0, and the student information table is shown in Table 1.

4.4.2. Menu Form. The menu table stores the full name (package name and class name) of the menu and the corresponding functional component of the menu in the database. When the user sends the menu name or menu number, the server will get the full name of the component corresponding to the menu and get the instance of the com-

ponent class through reflection. The menu E-R diagram is shown in Figure 10.

According to the menu E-R diagram, design the main fields of the menu table, as shown in Table 2.

Of course, according to the functional requirements of the system, the system database tables also include course schedules, message tables, and notification tables, which are similar to the above two tables, so I will not explain them in detail here.

4.5. System Category

4.5.1. Communication Interface Class. The WeChat public platform is different from the ordinary website system. The front page and server side of the website system have many communication interfaces. The user's request is directly sent to the corresponding interface, and each interface only handles specific types of requests. The data interaction between the WeChat server and the server of the microteaching service platform is completed through the Servlet class of Weixin.java, and all codes of the server of the microteaching service platform are executed in this class. Since the user's request is no longer sent to the corresponding interface, the user's request needs to be judged in this interface, and the corresponding business logic is called to complete the response. This class defines two methods doGet() and doPost() [30–32].

(1) *doGet()*. The doGet() method is only called once when the microteaching service platform is accessed. The user sends various requests on the WeChat client, and the request is forwarded by the WeChat server to the URL corresponding to the server of the microteaching service platform. Therefore, it is necessary to bind the WeChat official account with the server of the microteaching service platform to enable users to teach from micro. The message sent by the official account of the service platform is delivered to the URL of the server.

After filling in the url and token, click Submit, and then, call the checkSignature() verification function to verify. This function first sorts the three parameters of token, time, and nonce, then concatenates the three parameter strings into a string for sha1 encryption, and finally compares the encrypted string with the signature to verify whether the two strings are the same. If the verification is successful, the echostr string is returned as it is, indicating that the access is successful.

(2) *doPost()*. After the binding is successful, the request sent by the application will call the doPost method. After the connection is successful, the server will use the POST method to forward the user's request to the server of the digital teaching service platform. This class calls the doPost method after receiving the request. After the system server receives the message, it first parses the message. This class encapsulates the request analysis into a subclass of WeixinRequest and then uses the HandleChainManager class to process the request. The request processing is a chain processing

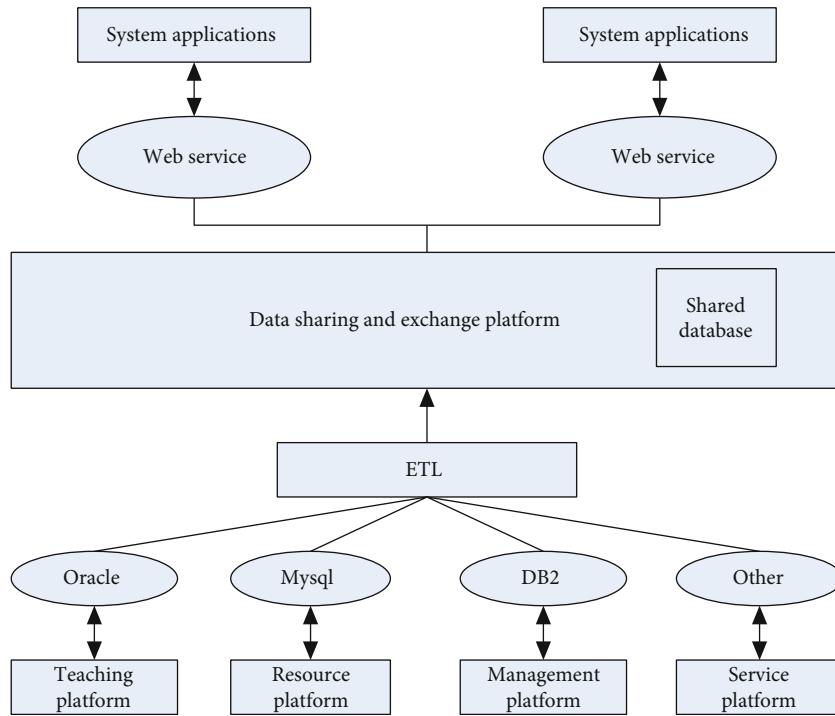


FIGURE 7: The overall architecture of the data sharing and exchange platform.

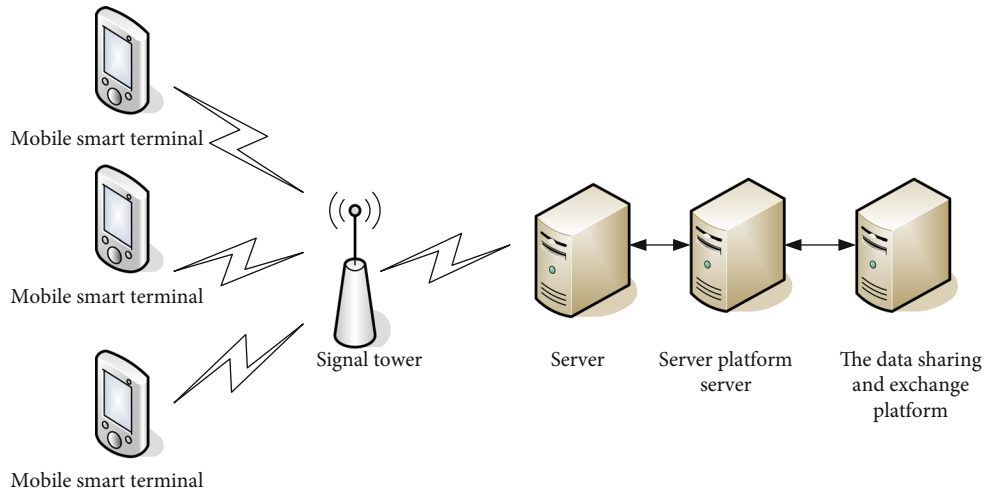


FIGURE 8: System architecture.

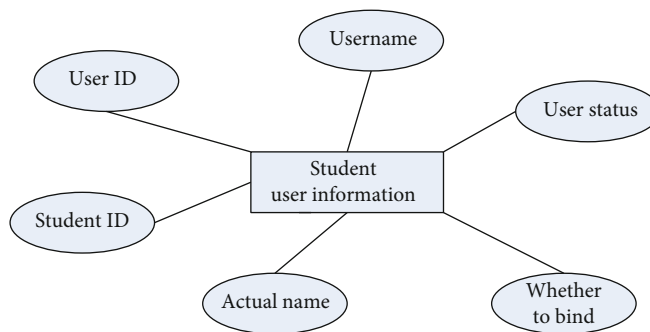


FIGURE 9: Student user information E-R diagram.

TABLE 1: Student information table.

Chinese name	Field name	Type of data	Length	Is it empty	Remarks
User ID	Id	Int	20	No	Primary key
Username	Username	varChar	30	Null	
User status	Status	Int	20	Null	
Bind	Bind	Boolean	1	Null	
Real name	RealName	varChar	20	Null	
Student ID	StuNo	varChar	20	Null	

process, matching the requests one by one, and finally get WeixinResponse response. Each function of the system corresponds to a WeixinComponent class. During the request processing, if the message sent by the user matches the system menu, the function component corresponding to the menu is called through the WeixinComponentUtil class in the request processing class to get a WeixinResponse response. The communication interface class finally packs it into the corresponding XML format according to the WeixinResponse type and returns it to the server.

4.5.2. Request Message Class

(1) *Request Message*. When the user sends a message to the system, the server fills the POST message in the URL according to the XML format. First, it parses the XML data and stores the data in the XML into the corresponding entity class. The system allows users to send many types of messages and different formats, so the request message interface is defined, interface-oriented programming technology is used to encapsulate text, location, picture, link, and event messages, and each type of WeChat message is encapsulated into JavaBean. When in use, we use the polymorphism of the object-oriented language to create WeChat request message objects. The WeixinRequest interface defines the getReq_type() method for obtaining the message type, the getFromUserName() method for obtaining the message sender account, and the getToUserName() method for obtaining the message receiver account. Text request (TextRequest), location request (LocationRequest), image request (ImageRequest), link request (LinkRequest), and event request (EventRequest) all implement the WeixinRequest interface, which overrides the methods in the interface to achieve the corresponding functions.

(2) *Response Message*. For the request sent by WeChat using the POST method, the system needs to package the response content into a specified XML format before it can be forwarded to the user through the WeChat server. WeChat can respond to users with many types of messages. These message formats are different. Therefore, we define the WeChat response message interface, adopt interface-oriented programming technology, use different subclasses to encapsulate different WeChat response messages, use object-oriented polymorphism of creating a WeChat response message object, and finally packaged into the corresponding XML format according to the message type.

The WeixinResponse interface defines the getReq_type() method to obtain the message type. The text response (TextResponse), the music response (LinkResponse), and the graphic response (NewsResponse) all implement the WeixinResponse interface, which overrides the methods in the interface to implement the corresponding functions. When the response is packaged, the response type is obtained through the getReq_type() method, and the response type is packaged into the corresponding XML format.

4.5.3. *Message Analysis Class*. When the user uses the client to send a message to the system, the server will package the message into an XML data packet and forward it to the digital teaching service platform server. After the digital teaching service platform server receives the data, it first parses the XML message and stores it in the corresponding entity class. There are two main methods of parsing XML: DOM parsing and SAX parsing. DOM is based on the XML document tree structure, and SAX is based on event flow. Load the entire document tree into memory when parsing. The platform stipulates that the message interacting with the customer service server is an XML string in a specified format. The message is a simple tree structure and will not take up a lot of memory when all loaded, so the open source framework dom4j is chosen to parse the XML data.

First, the WeixinRequest interface is declared, which is used to store the analysis results. Second, obtain the input stream through the getReader() method, convert the original XML format message into a String type string, and pass the String type string obtained to the parseText(msg) method to get the entire DOM, including the root node of XML and all Child node. Again, call the getRootElement() method through the doc object to get the root node object rootElt, and pass the child node name to the elementTxt() method to get the content in the child node. Finally, according to msgType, the message type, the subclass of the WeixinRequest interface is instantiated, the input stream is closed, and the WeixinRequest interface instance is returned to complete a parsing.

4.5.4. *Request Processing*. Request processing categories include notification requests, event requests, text requests, keyword requests, and menu requests. When designing the system, different Servlet classes are generally used to receive different requests. This system uses the same Servlet class to receive and respond to user requests forwarded by the client server. Therefore, in the Servlet class, the request type must be determined, and different requests must be processed

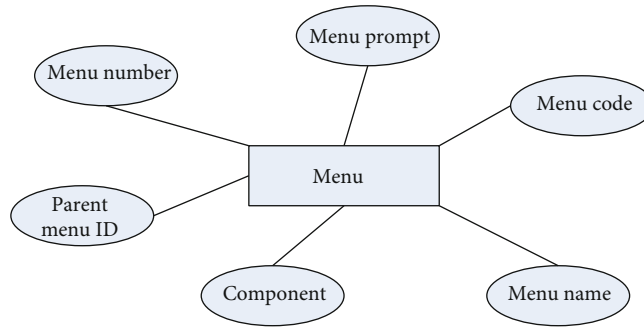


FIGURE 10: Menu E-R diagram.

TABLE 2: Menu table.

Chinese name	Field name	Type of data	Length	Is it empty	Remarks
Menu number	Id	Int	20	No	Primary key
Menu prompt	Contents	varChar	50	Null	
Menu code	Menucode	Int	20	Null	
Menu name	Menuname	varChar	30	Null	
Component	Outurl	varChar	20	Null	
Parent menu ID	Pid	Int	20	Null	Foreign key

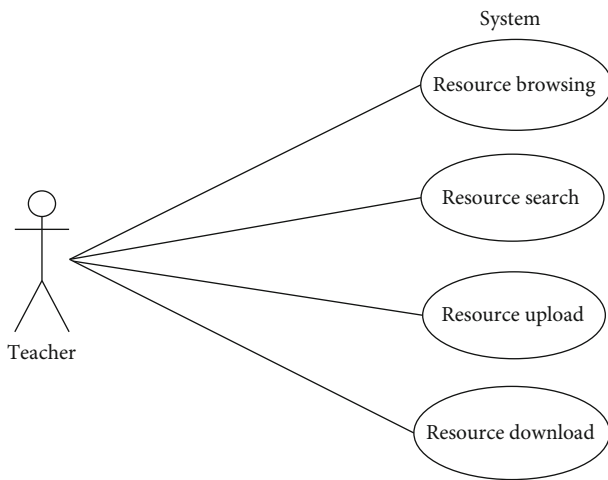


FIGURE 11: Test case diagram.

TABLE 3: Teacher login system test.

Teacher ID	Password	Result
Currently logged in	Correct	Correct
Not currently logged in	Correct	Error
Not currently logged in	Error	Error
Garbled	Error	Error

differently to receive responses. In order to improve system performance, increase code reuse rate, and facilitate subsequent maintenance; this process is encapsulated in the system, and the request processor chain is used to match the request types one by one in the order in the configuration

file wechat.cfg.xml. Then, if there is a match, the corresponding processing is performed. After the processing is completed, a response is obtained and the request ends.

4.5.5. *Functional Component Class.* Information inquiry is the core of the digital teaching service platform, which includes functions such as course inquiry, grade inquiry, teacher inquiry, examination inquiry, and borrowing inquiry. A set of common component interfaces is defined in the cn.edu.hnust.weixin.components package, and each function corresponds to a component. To write any component, it is necessary to implement the universal component interface and realize the universal processing method of the component.

The system defines subinterfaces for teachers and student users and also defines a public interface for public content. The functional components that inherit this interface can be used without logging in. The three interfaces all inherit the WeixinComponent interface, namely, the teacher component interface TeacherWeixinComponet, the student component interface StudentWeixinComponet, and the public component interface PublicWeixinComponet.

In the corresponding components, if it is a student functional component, it implements the StudentWeixinComponet interface and so on. For public components, it inherits the PublicWeixinComponet interface. The handle method in the component interface must be overwritten in the functional component class.

Component classes have different ways of obtaining data, which can be roughly divided into three categories: calling the Web Service interface, connecting to a shared database to query views, and crawling web pages. The following are examples of the implementation methods of these

TABLE 4: Part of the system function test content.

Function	Content	Result
User management	Register; login	Success
	Change password	Success
	Information query	Success
Course operation	Course inquiry	Success
	Course addition	Success

three types of components. The components implemented by the Web Service interface take the borrowing query component as an example, the components searched in the shared database view are taken as an example for the course score query, and the web page component is used for CET score query as an example.

(1) *Course Score Query*. When querying the course scores, students enter the course name, and the system server accesses the shared database to find the corresponding score information in the student score table according to the student's student ID and the entered course name. Each time the course score query component is called, there are two interactions. The first time the component is called, the query format is returned, that is, the course name and the query content are returned only when the user enters the correct query format for the second time.

(2) *Borrowing Inquiry*. The borrowing query component is implemented by using the Web Service interface. A functional component implemented by using the Web Service interface must first import the JAR file of the interface into the project, and then implement the interface according to the interface document to obtain a method to obtain interface data, and finally call it in the component's handle() method. The two entity classes defined in the borrowing query component package, LibraryBookInfo and ReaderInfo, correspond to book information and reader information. The borrowing query component obtains the reader information and borrowing information of teachers and students by passing in student ID or faculty ID.

(3) *CET Score Query*. Since the educational administration management system does not provide a query interface for the fourth and sixth levels, the score data of the fourth and sixth levels are obtained from external websites. The source of the score data for the fourth and sixth grades is <http://www.chsi.com.cn/cet/>. Through HttpClient simulated login to obtain all the html information of the response webpage, use Jsoup to grab the relevant part of the fourth and sixth grades of the information data. The fourth- and sixth-level query requires two interactions. The first time you enter the CET score query menu, the query format candidate number+name is prompted. The second time the user enters the candidate number+name format, the system server returns the fourth- and sixth-level results.

4.5.6. *Component Call Class*. Component calls are all done using the WeixinComponentUtil class. A static method is

defined in the ComponentUtil class. The menu object is passed into the method. The menu object is used to obtain the complete class name of the menu corresponding to the component plus the package name, and then, the component class is obtained through reflection. For instance, after getting the component class instance, you need to determine the user permissions.

If it is an object of TeacherWeixinComponet, StudentWeixinComponet subclass, call the getBind() method to get the user binding status through the user name; if it is a bound user, you can use the component; if it is not bound, the link message "Bind now" will be returned; click Bind now to jump to the binding interface. If it is a subclass object of the public component interface PublicWxComponet, the component is called directly.

Each menu item has a corresponding complete class name plus package name, and the corresponding class can be found and loaded through reflection. If you want to add a new query function, you only need to put the corresponding query component into a jar package and put it under the classpath of the server, and add the corresponding menu item to add a new query function.

5. Testing and Analysis of Mobile Information Systems

5.1. Test Environment

5.1.1. *Server*. Server operating system: Windows 2008 Server
Server database system: SQL server 2010
Web server configuration: Tomcat7.0
Database: MySQL 5.1
Memory: 2 G

5.1.2. *Client*. Phone model: Xiaomi 10
System version: Android 11.0
Memory: 8 G

5.2. *Test Content*. This article has carried on the unit test to the system, also called the module test. The system test uses a combination of black box testing and white box testing for system testing. The main difference between black box testing and white box testing is whether to consider internal algorithms and whether to consider internal structures. The test case diagram is shown in Figure 11.

5.2.1. *Black Box Test*. In black box testing regardless of the internal algorithm and structure of the system, it only tests the function and operation of the system to check whether the system functions are consistent with expectations and conform to the functional design of the system. Therefore, black box testing is also called functional testing.

The teacher login system test is shown in Table 3.

The specific operation items and results of the system function test content are shown in Table 4.

5.2.2. *White Box Testing*. White box testing is a comprehensive inspection of the system processing details. It allows programmers to use the internal logical structure information of the system and test whether the logical path

TABLE 5: User management test.

Function	Test results	Conclusion
Turn on XXX as system user	Enter the system to open new users	The result is consistent
Admin user login password modify	Password can be changed	The result is consistent
Admin user settings review authority	Can set different usage permissions for users	The result is consistent

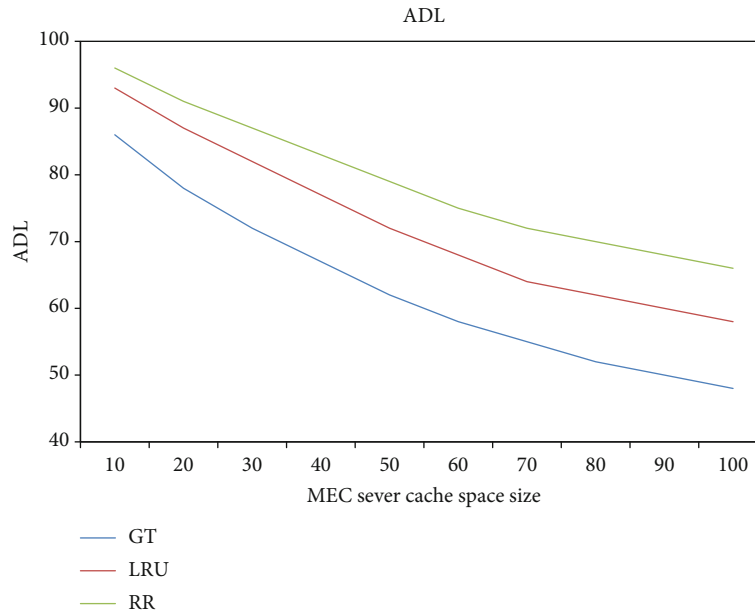


FIGURE 12: MEC sever cache space size.

conditions in the system are consistent with expectations. Therefore, the white box test is also called structural test.

The user management test is shown in Table 5.

5.3. Test Analysis. It can be seen from Figure 12 that the average content transmission delay decreases as the server cache space increases. Because the cache space is increased, the local cache can store more content so that more user requests can be directly served by the local cooperative cache domain. Experimental data shows that when the storage space ranges from 50 to 100, the system combines with the MEC server under the LRU and RR mechanisms, and the ADL is lower by about 14%~22% and 21%~31%, respectively.

After the above test, it can be determined that the system can basically operate normally. The system can correctly perform the main functions of this design and can correctly handle abnormal conditions within the expected range. After testing, it can be found that the college digital teaching information system has a simple interface, simple operation, and functions that meet expectations, which can meet the needs of digital teaching. In this chapter, black box and white box tests are carried out on each functional module of the digital teaching system in colleges and universities. The test results show that the system functions can be accessed normally, which reduces the transmission delay and improves the efficiency of digital teaching in colleges and universities.

6. Conclusions

Mobile information system and edge computing technology have outstanding construction for the digital teaching of college English. This article combines edge computing and cloud computing technology to design a specific information system, integrates various information resources of the school, analyzes the specific work process, and researches in the digital campus. Digital teaching platform is established. This system can realize digital teaching, collect and integrate teaching information, provide support for modern education, and realize resource sharing. The digital teaching information system includes basic teaching information, information browsing functions, course inquiry, campus map, and campus recruitment. Not only that, due to the rapid growth of smart mobile terminal devices such as smart phones, smart phone clients have been designed to facilitate users to access digital teaching anytime and anywhere. The mobile information system has greatly improved the efficiency of digital teaching management in universities and accelerated the realization of digital English teaching in universities.

Data Availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The author declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

References

- [1] V. Sathiyamoorthi, P. Keerthika, P. Suresh, Z. (. J.). Zhang, A. P. Rao, and K. Logeswaran, "Adaptive fault tolerant resource allocation scheme for cloud computing environments," *Journal of Organizational and End User Computing*, vol. 33, no. 5, pp. 135–152, 2021.
- [2] M. F. Tefek, H. Uğuz, and M. Güçyetmez, "A new hybrid gravitational search–teaching–learning-based optimization method for energy demand estimation of Turkey," *Neural Computing and Applications*, vol. 31, no. 7, pp. 2939–2954, 2019.
- [3] C.-A. Yuan, Z. Zhong, C. Lei, X. Zhu, and R. Hu, "Adaptive reverse graph learning for robust subspace learning," *Information Processing and Management*, vol. 58, no. 6, article 102733, 2021.
- [4] H. F. Farahani, J. Aghaei, and F. Rashidi, "Optimal power flow of HVDC system using teaching–learning-based optimization algorithm," *Neural Computing and Applications*, vol. 30, no. 12, pp. 3781–3789, 2018.
- [5] R. Kanniga Devi, M. Gurusamy, and P. Vijayakumar, "An efficient cloud data center allocation to the source of requests," *Journal of Organizational and End User Computing*, vol. 32, no. 3, pp. 23–36, 2020.
- [6] V. Mathew, R. K. Sitaraman, and P. Shenoy, "Energy-aware load balancing in content delivery networks," *Infocom Proceedings IEEE*, vol. 131, no. 5, pp. 954–962, 2017.
- [7] F. Jiayi, "China's artificial intelligence industry chain "shows mountains and dew"," *Decision Exploration (first half of the month)*, vol. 11, pp. 38–40, 2017.
- [8] W. Kazimierski and M. Wlodarczyk-Sielicka, "Technology of spatial data geometrical simplification in maritime mobile information system for coastal waters," *Polish Maritime Research*, vol. 23, no. 3, pp. 3–12, 2016.
- [9] F. Mata, M. Torres-Ruiz, G. Guzmán et al., "A mobile information system based on crowd-sensed and official crime data for finding safe routes: a case study of Mexico City," *Mobile Information Systems*, vol. 2016, no. part 1, 11 pages, 2016.
- [10] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: vision and challenges," *Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.
- [11] M. Satyanarayanan, "The emergence of edge computing," *Computer*, vol. 50, no. 1, pp. 30–39, 2017.
- [12] W. Shi and S. Dustdar, "The promise of edge computing," *Computer*, vol. 49, no. 5, pp. 78–81, 2016.
- [13] M. Dias de Assunção, A. da Silva Veith, and R. Buyya, "Distributed data stream processing and edge computing: a survey on resource elasticity and future directions," *Journal of Network & Computer Applications*, vol. 103, no. FEB., pp. 1–17, 2018.
- [14] O. Maher and E. Sitnikova, "A trustworthy learning technique for securing industrial Internet of things systems," *Journal of Intelligent Systems and Internet of Things*, vol. 5, no. 1, pp. 33–48, 2021.
- [15] X. Ma, C. Lin, H. Zhang, and J. Liu, "Energy-aware computation offloading of IoT sensors in cloudlet-based mobile edge computing," *Sensors*, vol. 18, no. 6, p. 1945, 2018.
- [16] S. Kim, "One-on-one contract game-based dynamic virtual machine migration scheme for mobile edge computing," *Transactions on Emerging Telecommunications Technologies*, vol. 29, no. 1, p. 3204, 2018.
- [17] T. Taleb, K. Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On multi-access edge computing: a survey of the emerging 5G network edge cloud architecture and orchestration," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1657–1681, 2017.
- [18] Y. Wang, M. Sheng, X. Wang, L. Wang, and J. Li, "Mobile-edge computing: partial computation offloading using dynamic voltage scaling," *IEEE Transactions on Communications*, vol. 64, no. 10, pp. 1–4282, 2016.
- [19] S. Jeong, O. Simeone, and J. Kang, "Mobile edge computing via a UAV-mounted cloudlet: optimization of bit allocation and path planning," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 3, pp. 2049–2063, 2018.
- [20] Y. Mao, J. Zhang, S. H. Song, and K. B. Letaief, "Stochastic joint radio and computational resource management for multi-user mobile-edge computing systems," *IEEE Transactions on Wireless Communications*, vol. 16, no. 9, pp. 5994–6009, 2017.
- [21] N. D. Tung, L. L. Bao, and B. Vijay, "Price-based resource allocation for edge computing: a market equilibrium approach," *IEEE Transactions on Cloud Computing*, vol. PP, pp. 1–1, 2018.
- [22] K. Zhang, Y. Mao, S. Leng, Y. He, and Y. ZHANG, "Mobile-edge computing for vehicular networks: a promising network paradigm with predictive off-loading," *IEEE Vehicular Technology Magazine*, vol. 12, no. 2, pp. 36–44, 2017.
- [23] H. Li, K. Ota, and M. Dong, "Learning IoT in edge: deep learning for the Internet of things with edge computing," *IEEE Network*, vol. 32, no. 1, pp. 96–101, 2018.
- [24] D. Sabella, A. Vaillant, P. Kuure, U. Rauschenbach, and F. Giust, "Mobile-edge computing architecture: the role of MEC in the Internet of things," *IEEE Consumer Electronics Magazine*, vol. 5, no. 4, pp. 84–91, 2016.
- [25] A. C. Baktir, A. Ozgovde, and C. Ersoy, "How can edge computing benefit from software-defined networking: a survey, use cases & future directions," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2359–2391, 2017.
- [26] M. Chen and Y. Hao, "Task offloading for mobile edge computing in software defined ultra-dense network," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 3, pp. 587–597, 2018.
- [27] F. Karakoyun and M. Yapc, "Use of digital storytelling in biology teaching," *Universal Journal of Educational Research*, vol. 4, no. 4, pp. 895–903, 2016.
- [28] M. Claro, A. Salinas, T. Cabello-Hutt et al., "Teaching in a digital environment (TIDE): defining and measuring teachers' capacity to develop students' digital information and communication skills," *Computers & Education*, vol. 121, no. JUN., pp. 162–174, 2018.
- [29] V. Jain, M. Gupta, N. Joshi, A. Mishra, and V. Bansal, "E-College: an aid for E-learning systems," *Fusion: Practice and Applications*, vol. 3, no. 2, pp. 66–73, 2021.
- [30] M. Koehler, S. Greenhalgh, J. Rosenberg, and S. Keenan, "What the tech is going on with teachers' digital teaching portfolios? Using the TPACK framework to analyze teachers'

technological understanding,” *Journal of Technology & Teacher Education*, vol. 25, pp. 31–59, 2017.

- [31] J. Breakstone, S. McGrew, M. Smith, T. Ortega, and S. Wineburg, “Why we need a new approach to teaching digital literacy,” *Phi Delta Kappan*, vol. 99, no. 6, pp. 27–32, 2018.
- [32] M. Buchbender, M. Maser, F. W. Neukam, M. R. Kesting, S. Attia, and C. M. Schmitt, “Kobra surgery simulator—a possibility to improve digital teaching? A case-control study,” *International Journal of Environmental Research and Public Health*, vol. 18, no. 4, p. 1827, 2021.