

Building of Online Evaluation System Based on Socket Protocol

Peng Jiang¹, Kexin Yan^{2,*}, Haijian Chen³ and Hai Sun⁴

¹ Jingan Branch Campus, Shanghai Open University,
Shanghai 200040, China
jzhpmail@163.com

² Management School, Shanghai University of International Business and Economics,
Shanghai 201620, China
yqx980219@163.com

³ Shanghai Academic Credit Transfer and Accumulation Bank for Lifelong Education,
Shanghai 200433, China
xochj@sou.edu.cn

⁴ School of Management, Fudan University
Shanghai 200433, China
sunhai@fudan.edu.cn

Abstract. As an important part of the evaluation reform, online evaluation system can effectively improve the efficiency of evaluation work, which has been paid attention by teaching institutions. The online evaluation system needs to support the safe and stable transmission of information between the client and the server, and socket protocol establishes the connection through the listening port, which can easily carry out the message transmission and process control. Because it can well meet the construction requirements of online evaluation system, it is applied in our study. The building of online evaluation system based on socket protocol includes the function design of students and teachers, data flow design, evaluation difficulty grading design and system implementation. The system uses Java language and MVC mode for development, which has good scalability and platform-independence. It realizes the paperless examination process and greatly reduces the workload of teachers. The contribution of this paper is mainly reflected in two aspects. One is to explore the construction of an online evaluation system based on the socket protocol, and it provide an Asynchronous IO technical solution for the network communication between the student and the server, which provides a reference for the development of similar systems. The second is to give the realization method of the difficulty classification of the evaluation, and classify the difficulty of the test questions, which lays the foundation for carrying out personalized testing and evaluation.

Keywords: online evaluation system; socket protocol; MVC mode; Asynchronous IO.

* Corresponding author

1. Introduction

With the promotion of the Internet and the development of educational information technology, online education has emerged. This new teaching method has the advantage of unlimited time and place [21], and it is welcomed by more and more learners. At the same time, online evaluation system has been rapidly promoted in many teaching institutions because of its convenient test method and paperless characteristics [15]. The research of online evaluation system has also become one of the hot spots of online education research, which has attracted the attention of many scholars [1][2][10][17]. However, because the online evaluation system needs to consider not only the concurrent access of users, but also the stability of data transmission and the security of the system [16], its construction presents a certain degree of difficulty, which is higher than that of the common management information system.

In order to meet the requirements of concurrency and security of online evaluation system, this paper presents a construction method of online evaluation system based on socket communication protocol [3][14]. The system completes the communication between terminal and server through asynchronous IO mode, which effectively ensures the safe transmission of data. In addition, the system is realized by adopting the MVC design pattern based on the J2EE framework [13][22][23], and it has good cross-platform and adaptability.

2. Related Research

With the advancement of teaching reform and the development of Internet, online evaluation system came into being. For example, the University of Valladolid system, which was born in 1995, it undertook the service of ACM International Undergraduate Program Design Competition [19]. Since then, similar systems have been emerging and their functions have become more and more perfect. In 2003, the blackboard platform launched for the teaching of Chinese colleges and universities has the functions of uploading courseware, correcting homework, testing and scoring. It plays an important role in the teaching of colleges and universities and is welcomed by many colleges and universities. Since then, the online evaluation system has obtained a good opportunity for development, and many online education testing platforms have appeared at home and abroad, such as Coursera, Udacity, edX and so on [6].

Because online evaluation system has many advantages, some colleges and universities have developed their own online examination system, which can realize the random combination of test papers and real-time performance presentation. For example, Tsinghua University and East China Normal University began to try to use Internet technology to change students' learning and examination methods. However, the original technology of the evaluation system is not perfect, and the function of the system is limited to simple operation and examination. Since then, more and more scholars began to study the design of online evaluation system, and they think about how to make this system better meet the needs of users. Kang et al. (2004) used J2EE technology to design the evaluation system. Their research gave the method of selecting test questions to form a test and showed the advantages of online classroom. The

practice results show that J2EE can support distributed applications and improve the evaluation efficiency [7]. Wang (2014) developed an online examination system based on IDC, which realized the real paperless online examination [20]. Although the above system has promoted the research and development of online examination to a certain extent, it is still not complete in function and lack of strong universality. In order to solve these problems, scholars have carried out more in-depth research on online evaluation system. Chen (2020) took the design of "data structure" course as an example. Their research is based on B/S mode, and used the SpringBoot framework of J2EE to develop the online evaluation system. The system has passed the stress test and stability test, which proves that the system can effectively improve the efficiency of online evaluation [4]. Zhong et al. (2020) analyzed the requirements of online evaluation system, which including front desk service module, background management module and evaluation module, and their study introduced the whole process of online evaluation system [29]. Zhang et al. (2018) designed an online evaluation system based on struts and Ajax technology to solve the problem of low openness of the current online evaluation system. The evaluation system has good openness and can be used to assist course teaching [26].

3. Functional Design

The development of the system follows the idea of multi-layer architecture [18], and it uses configuration files to save general configuration information. The reading and parsing operation of information is performed by the reading tool class Config class. Users can invoke various commands in the program to change the parameters and attributes of each component. The system interface is implemented by JFrame window class, JFrame can be regarded as a container, and all components used in the interface can be put into the container [25][28]. The system is divided into many functional modules, such as teachers' questions, changing questions, arranging examinations, viewing results, analyzing results, students previewing test papers, participating in examinations, querying results and so on.

The whole server uses socket communication protocol. When the client of the terminal calls the ExamServiceAgentImpl interface to access the service ExamServiceServer, the student side will pass the command name, parameter type, parameter value and an empty Sid to the server through an encapsulation class request. At this time, the server will assign a unique Sid to each user whose Sid is empty, and call ExamServiceImpl to respond to the request. The response of the above request is realized by reflection, so an encapsulation class response is returned to the student side, which contains the return value of the command and the new Sid. After that, all requests transmitted by the user will be bound to this SID, and the server will find the corresponding function implementation class of the core business according to the SID to implement the corresponding request and return the data. The structure diagram of the server is shown in Figure 1.

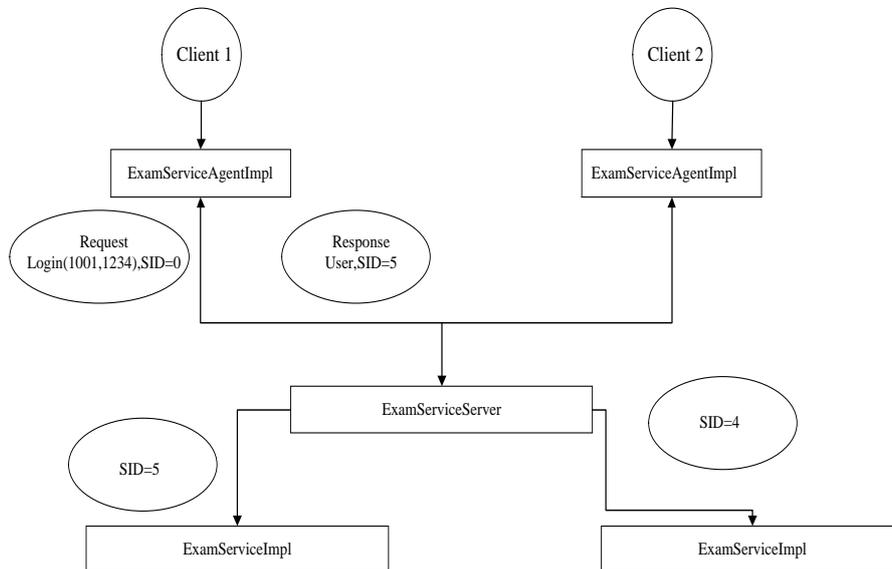


Fig. 1. The structure diagram of Server-side

3.1. Functional Design for Student

The main operations of the student side include login, preview, answer and submit. First, students start the student terminal, and input the user’s name and password in the authentication interface, and enter the student terminal interface to prepare for the exam after the verification is successful. At this time, students can only preview the test paper. When the test time arrives, students can click the start button to answer the questions. In the whole test process, students must complete the single choice questions, multiple choice questions, judgment questions, fill in the blanks and question and answer questions within the specified time, and then submit the test paper. After the whole test, students will get the test scores. If the test paper is not submitted in the specified time, the system will force students to submit the test paper.

The entire student terminal adopts the MVC design pattern [9][27]. When the user makes a request from each Frame every time, various commands in ExamService are called through Client Context, and the function is realized by ExamServiceImpl class. The EntityContext class is the tool class of question bank and student information import, which runs automatically when the program starts. After the user logs in successfully, it reads the properties setting file through the Config class before the exam starts, and imports test questions and student information into the program. The student structure is shown in Figure 2.

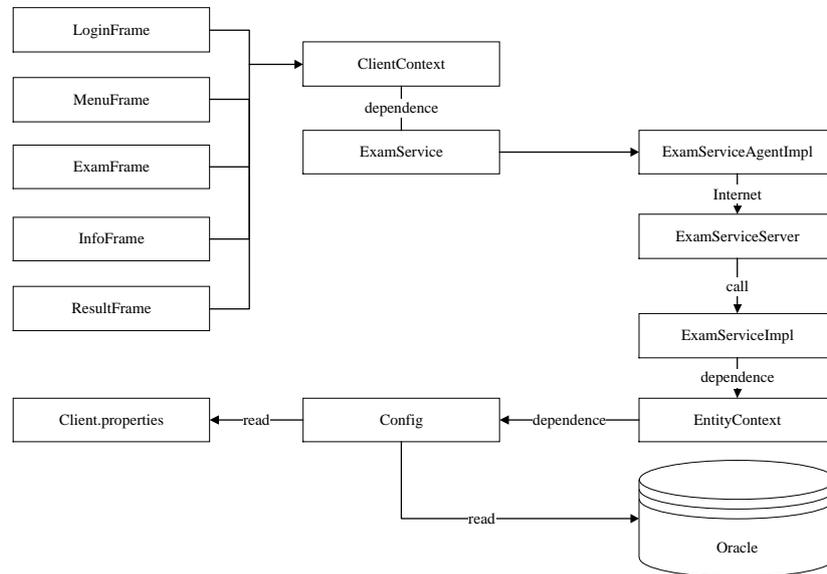


Fig. 2. The structure diagram for student

3.2. Functional Design for Teacher

The main operations of the teacher side include login, test paper making, examination arrangement and score inquiry. The teacher enters the user's name and password in the authentication interface, and enters the teacher interface once the authentication is successful. Among, test paper making includes adding test questions, importing test question set in batch and modifying test questions. Through the above functions to complete the production of the whole set of test papers and add the test questions to the test database.

Examination arrangement can set examination information such as test paper name, test time, the start time, number of questions and total score of test paper. It can save this information into the property's configuration file. After setting up the examination information, teachers will enter the interface of adding student information and add the student information involved in the examination into the user table of the database. Score inquiry allows teachers to query the results of all students. Teachers can also filter the query results according to the specified conditions if further requirements are required. In addition, teachers can enter the interface of grade analysis to check the correct situation of all students. In this interface, teachers can check the correct rate of each type and difficulty of questions to improve the teaching plan in the future. The basic flow chart is shown in Figure. 3.

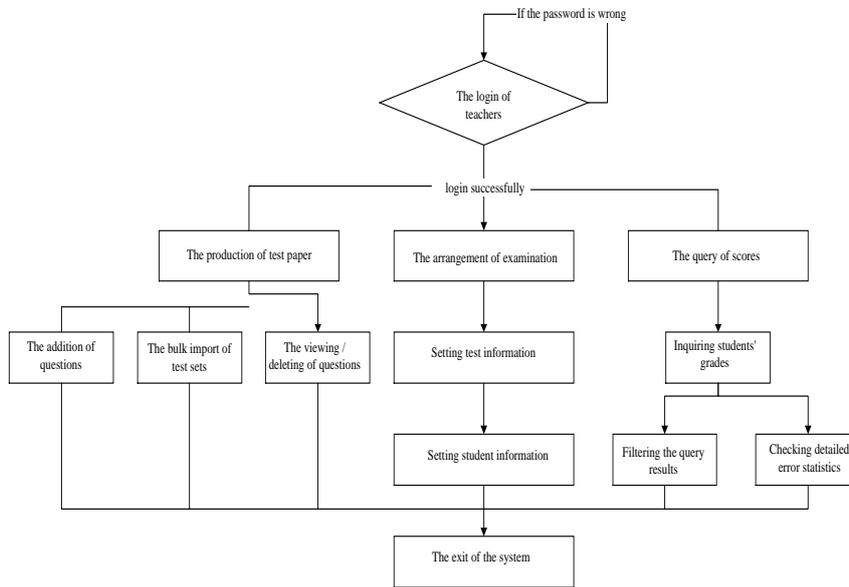


Fig. 3. The basic flow chart for teacher

The teacher side mainly completes the reading and editing of the database through the interface, and sets the examination information, database IP address and port, so it needs to have permission to operate the database and properties configuration file.

4. Analysis of Data Flow and Asynchronous IO

4.1. Analysis of Data Flow

The data flow chart is a tool to describe the data flow of the system, which uses the structured analysis method of top-down, layer-by-layer decomposition and step-by-step refinement [5][8][11]. At the same time, it uses a hierarchical DFD diagram to represent the transmission process between various data. The DFD diagram, also known as data flow diagram, is a tool to explain a series of processes such as system data input, output, storage and processing, which can help programmers to achieve effective cooperation between various modules [12][24].

In order to show the details of teachers and students in the whole system. On the basis of the top-level diagram, we decompose it from top to bottom and get the classification data flow chart of the online examination system. The classification data flow chart of the system is shown in Figure. 4.

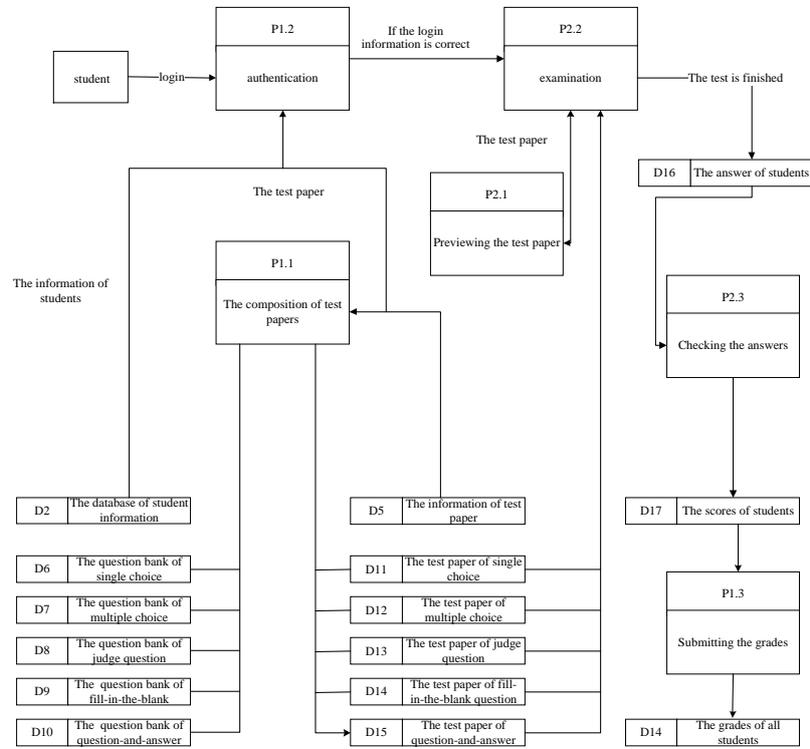


Fig. 4. The detailed flow chart for student and server

It can be seen from Figure. 4 that the online evaluation system is mainly divided into three sub-modules. That is, the student-side module, the server-side module and the teacher-side module.

4.2. Analysis of Asynchronous IO

Asynchronous IO is relative to synchronous io. The difference between asynchronous IO and synchronous IO is that when a thread performs IO operation, the operating system does not suspend the current thread operation. Instead, after the Input/Output instruction is executed, the operating system continues to let the current thread execute the next instruction. When the Input/Output operation is completed, the Input/Output thread will be notified through an event, and the thread will process the response event after receiving the notification. Specifically, the synchronous type allows multiple tasks to be completed through multiple threads, while the asynchronous type uses one thread to complete multiple tasks. When it encounters an I/O operation, it still lets the thread continue to execute other instructions, and notifies the thread to schedule a response event after the I/O is completed. The process of asynchronous IO in our study is shown in Figure 5.

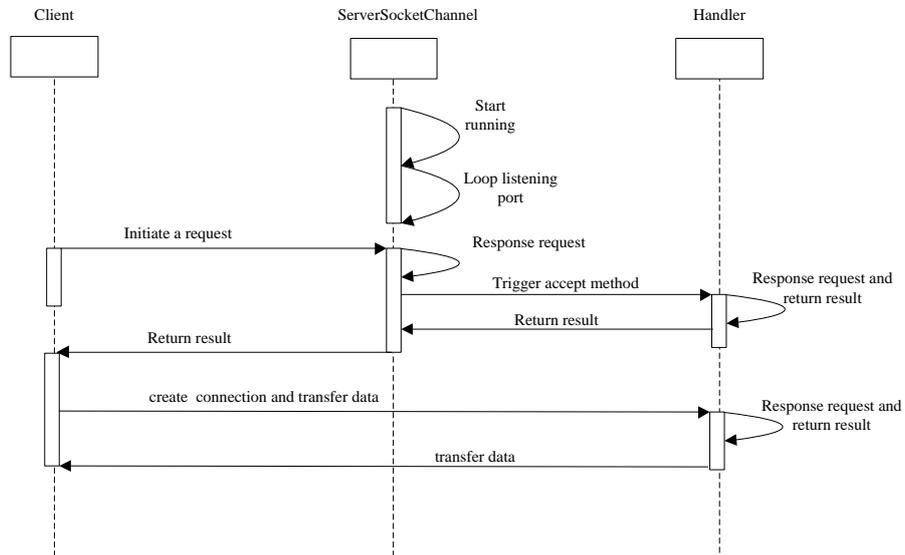


Fig. 5. The process of asynchronous IO

5. System Implementation

5.1. Implementation of Login Function

When the program starts, the system automatically starts the loadUsers method in the EntityContext to read all user names and passwords from the database. After the user entering the user's name and password in the login interface and clicking the login button, the system will call the GetId and GetPwd methods of the ClientContext (interface controller) to obtain the user's name and password entered by the user. Then, the user's name and password are introduced into getUserin the ExamServiceand compared with the set of users read from the database by the preprocessing utility class. If the User name and password are correct, the User data is returned. Finally, the User data is passed into the MenuFrame (student or teacher interface) and the interface is updated. At this point, the login of user is successful and the next operation can start. The whole process is shown in Figure. 6.

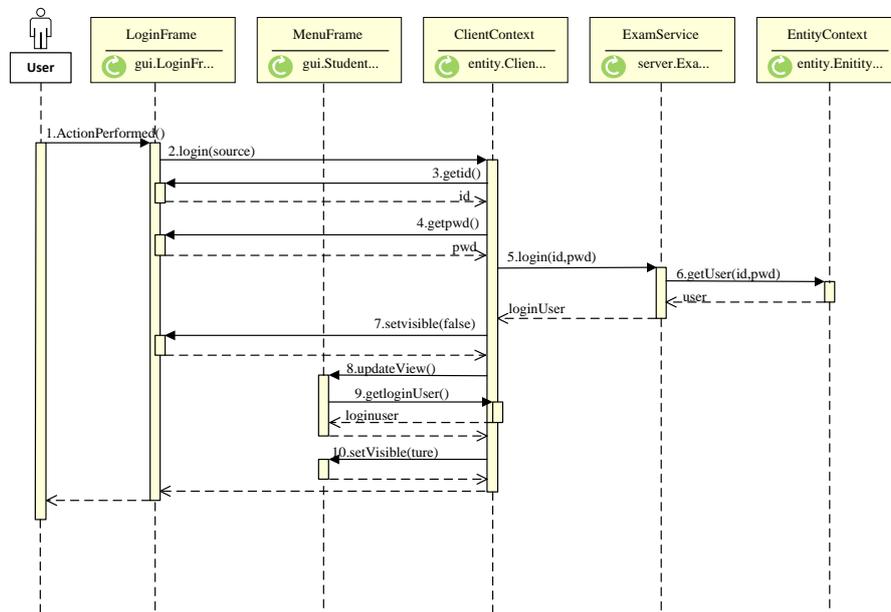


Fig. 6. The process of login function

Taking the login of student as an example, the procedure is executed as follows:

First, the EntityContext class is responsible for reading the user's name and password from the database and adding them into a collection. The steps are as follows:

- Step 1: Read student information from the database and add it to the array.
- Step 2: Read the information in Properties configuration file through the Config class and connect to the database.
- Step 3: Create SQL statement and pass it to the database to execute.
- Step 4: Get the query results of SQL.
- Step 5: Search all user information through SQL query statements and convert it to user data.
- Step 6: Add User data to student user's collection list.
- Step 7: Disconnect from database.

After that, the program obtains and transmits the user's name and password entered by the student through the ClientContext class, and the sample of key codes are follows.

```
public void login (JFrame source) {
    int id = loginFrame.getId();
    // Get the user's name from the login interface
    String pwd = loginFrame.getPwd();
    // Get the password from the login interface loginUser =
    service.studentlogin(id, pwd);
    // Check the user's name and password correctly through the
    relevant function of the core program
    studentMenuFrame.updateView();
    // Update user information on the student interface if successful
    studentMenuFrame.setVisible(true);
    // Enter the student interface
}
}
```

Finally, the ExamService implementation class is responsible for verifying that if the user's name password is correct. The steps are as follows:

- Step 1: Read the user's name entered by clientcontext.
- Step 2: The dialog box saying "no user" will be popped if the user's name is not found in the user collection.
- Step 3: If the user's name and password are input correctly, the user will be set as a valid user and the test paper generation operation is started.
- Step 4: Otherwise, it displays the dialog box with information of "the password is incorrect".

5.2. Implementation of Generating Test Paper Function

After successfully logging in, students will enter the main interface of students. Various loadQuestions methods of the EntityContext (preprocessor utility class) are automatically executed by the system. All questions are read from the question bank in the database, and then are added into a HashMap with Level or Score as the Key and the collection of this type of questions as Value according to the test paper requirements. They then call the buildPaper function in ExamServiceImpl (the core business implementation class) to establish the test paper for each type of topic, and use the getQuestions function to select the question from the HashMap and add it to a QuestionInfo. Each QuestionInfo contains all the attributes and the user's answer of the question.

Finally, all QuestionInfo were formed into a set "Paper" to form the test paper, which contains the information of the QuestionInfo and its number in the set (the index of the question in the set). The whole process is shown in Figure. 7.

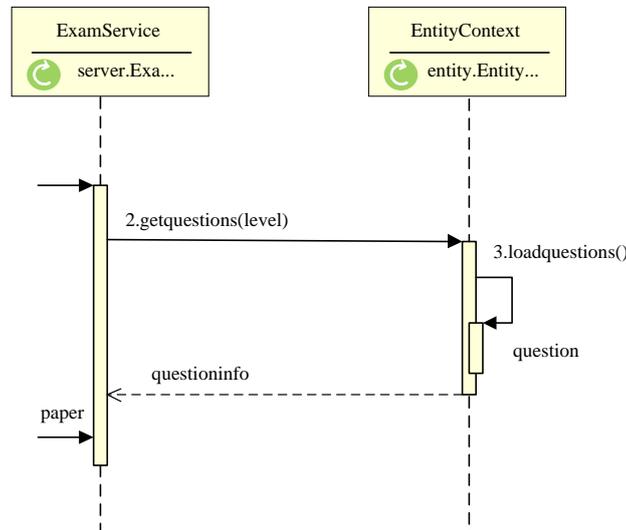


Fig. 7. The process of the function of reading question papers

Take the production of multiple-choice test papers as an example, the procedure is executed as follows:

First of all, to read the information of each question in the question bank through the EntityContext class. The steps are as follows:

- Step 1: Create a set of topics “Choice Questions” for multiple choice questions.
- Step 2: Read the information in the Properties profile through the Config class and connect it to the database.
- Step 3: Create SQL query statements and pass them to the database for execution.
- Step 4: Get the query result and convert it to an entity with multiple choice question.
- Step 5: The question is added to the choice levels (a HashMap based on the difficulty of the question), which will be used to complete the retrieval operation according to the requirements of the test paper in the future.
- Step 6: Close the connection to the database.

After reading the title, we form the test paper through the implementation class in the ExamService, and the sample of key codes are follows.

```
private List<ChoiceQuestionInfo> choicepaper = new
ArrayList<ChoiceQuestionInfo>();
// Create a set of multiple-choice questions with ChoiceQuestionInfo
private void buildChoicePaper() {
// Read the information in Properties settings file through the Config class
and connect it to the database
for (int level = 1; level <= 10; level++) {
// Through the outer cycle, each difficult topic can be selected
for (int k = 1; k <= count; k++) {
...// Establish a random number and get the questions from the
HashMap, It is through the inner loop to ensure a specified number of
questions can be obtained for each difficult topic, and then add the
question to the test set
}
}
... // Close the connection to the database
```

5.3. Implementation of Previewing Test Paper Function

When the user clicks the “preview test paper” button of the MenuFrame (student interface), the system sends the command of previewing test paper to the ClientContext (interface controller). Subsequently, the controller acquires all the test questions from the ExamService (business core logic) and converts them into strings. Finally, the string is passed into a TextArea of the preview interface (SearchQuestionFrame) and displayed for candidates to preview the test paper. The process is shown in Figure. 8.

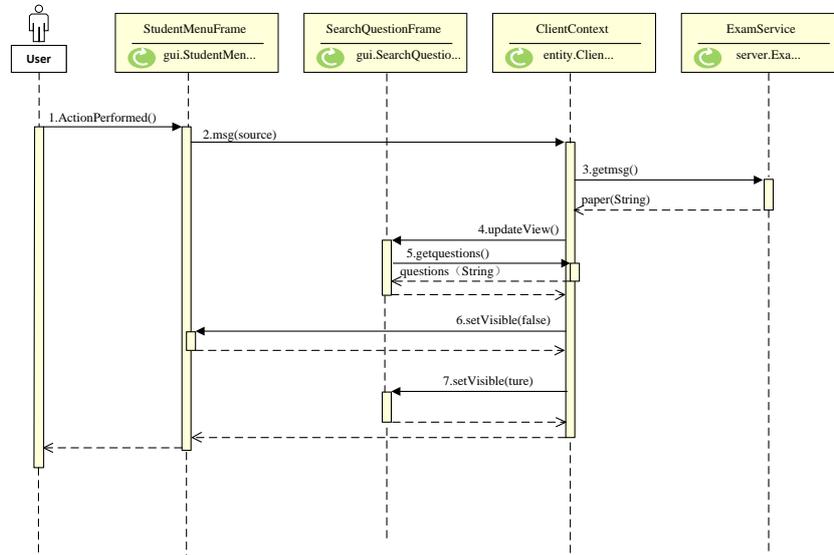


Fig. 8. The process of previewing test papers

5.4. Implementation of Examination Function

The whole examination is divided into five types: single-choice questions, multiple-choice questions, judgment questions, fill-in questions and question-answer questions. Although the type of data transmitted is different from the source of the information sent, the principle of starting the examination is the same.

When user clicks the button of starting test in the MenuFrame (the student interface), the system sends the command of starting test to ClientContext (the interface controller). If the specified test time is reached, the system sends the “start” command to ExamService (the business core program) to start the test. At this time, the timer starts, and the controller sends the getExamInfo command to the business core program to get the exam information. This information includes examination times, examination subjects and the number of questions.

Then the command of “getSingleChoiceQuestions” is sent to the business core program to get the test information of the first topic of the single topic. Finally, the first question and the examination information will be introduced into the single-choice examination interface and the examination interface will be updated, and the students will enter the single-choice interface to start answering questions.

The interface of the five types of questions has a common timer thread. When the timer is 0, the user's current information is saved and the user's test paper is forced to be submitted. The process is shown in Figure. 9.

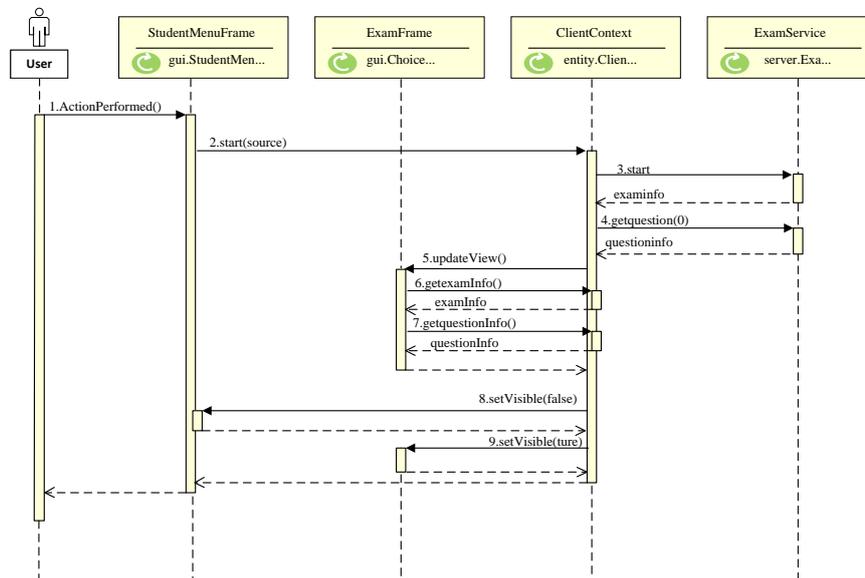


Fig. 9. The process of the function of starting examination

Taking the “start the single-topic examination” as an example, it will send a request to the core business by the controller, and the sample of key codes are follows.

```

public void startsinglechoice (JFrame source) {
    service.startsinglechoice();
    // Send a request of starting test
    singlechoiceQuestionInfo=service.getSingleChoiceQuestion(0);
    // Get the title information of the first single topic
    singlechoiceExamFrame.updateView();
    // Update the Test Interface of Single Topic
    ...
    // Leave the student interface and display the exam interface
    startTimer(); // Start timer thread
}
  
```

The steps of the timer thread are follows.

- Step 1: Get the exam time, the present time and the end time of the exam.
- Step 2: Start the thread to start the timer, calculate the remaining time of the exam and display it on the interface.
- Step 3: If the timer is timed out, the related operation is performed.

In the process of students participating in the online test, the system provides five types of test questions for students to choose according to the difficulty of the test questions. The difficulty classification of examination questions adopts fuzzy comprehensive evaluation method. In the fuzzy comprehensive evaluation, the weight will have a great influence on the final evaluation result. We use the expert estimation method to determine the weight. The steps of fuzzy comprehensive evaluation are as follows.

- Step 1: Determining the factor domain of evaluation object, $U = \{u_1, u_2, \dots, u_m\}$. there are m evaluation indexes, which indicate from which aspects we can judge and describe the evaluated object.

- Step 2: Determine the set of rating levels. The evaluation grade set is a set composed of various total evaluation results that the evaluator may make on the evaluated object, which is represented by V as follows. $V = \{V_1, V_2, \dots, V_n\}$. In fact, it is a division of the change interval of the evaluated object. Where v_i represents the i th evaluation result and n is the total number of evaluation results. The specific level can be described according to the test questions. For example, the difficulty degree of the test questions can be expressed by V , $V = \{\text{very difficult, relatively difficult, medium, relatively easy, very easy}\}$.
- Step 3: The single factor evaluation was carried out and the fuzzy relation matrix R was established. After constructing the fuzzy subset of the grade, the evaluated object should be quantified from each factor $u_i (i = 1, 2, \dots, m)$ that is to say, the membership degree of the evaluated object to each fuzzy subset of each grade from the single factor is determined, and then the fuzzy relation matrix is obtained as follows.

$$R = \begin{pmatrix} r_{11} & r_{12} & \Lambda & r_{1n} \\ r_{21} & r_{22} & \Lambda & r_{2n} \\ M & M & O & M \\ r_{m1} & r_{m2} & \Lambda & r_{mn} \end{pmatrix} \quad (1)$$

Where $r_{ij} (i = 1, 2, \dots, m; j = 1, 2, \dots, n)$ represents the membership degree of an evaluated object to the V_j -level fuzzy subset from the perspective of factor u_i . The performance of an evaluated object in a factor u_i is described by fuzzy vector $r_i = (r_{i1}, r_{i2}, \dots, r_{in})$. r_i is called single factor evaluation matrix, which can be regarded as a fuzzy relationship between factor set U and evaluation set V .

- Step 4: determine the fuzzy weight vector of evaluation factors. In order to reflect the importance of each factor, each factor u should be assigned a corresponding weight $a_i (i = 1, 2, \dots, m)$. a_i is usually required to conform to $a_i \geq 0, \sum a_i = 1$, and a_i represents the weight of the i th factor.

5.5. Implementation of Browsing the next Question Function

It is a process of submitting the answer and obtaining a new question that the student clicks on “previous question or next question”. For example, when a student clicks “the next question” button in the ExamFrame (exam interface), the program will save the answer of candidate into the test paper List of the ExamService (program core business) through the ClientContext (interface controller). At the same time, through the interface controller, the QuestionInfo of the next question will be returned to the test interface and the interface will be updated (if the question has been completed, the interface will be updated according to the answer selected by the user). At this point, the user can continue to complete the next question until the end of the exam. The whole process is shown in Figure. 10.

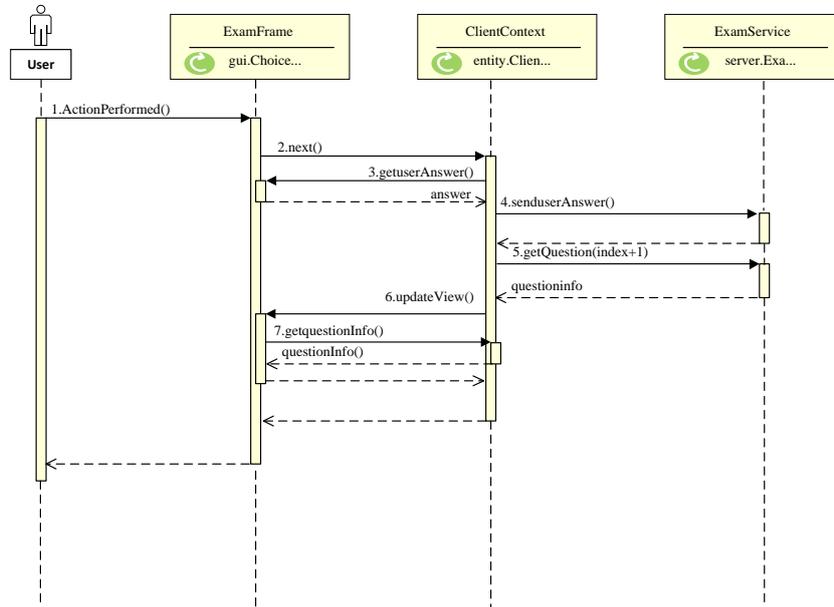


Fig. 10. The process of the function of the upper/lower topic

Next, taking the judgment question as an example, the process is shown as follows.

Firstly, the information of the next topic is obtained by “the next” command in the clientcontext class, and the sample of key codes are follows.

```

public void TFnext (JFrame source) {
    int index = tfQuestionInfo.getQuestionIndex();
    //Access to current title
    if (index + 1 == examInfo.getTFQuestionCount()) {
        return;}
    //If the title number is greater than the total title, no action is
    performed
    int answers = tfExamFrame.getUserAnswer();
    //Access to user answers
    service.sendUserTFAnswers(index, answers);
    //Transmit question numbers and user answers to business core
    programs
    tfQuestionInfo = service.getTFQuestion(index + 1); //Access to
    information of the next question
    tfExamFrame.updateView();
    //Update interface}
}
  
```

After that, the interface will be updated through the ExamMenu with the following steps.

- Step 1: Update the topic information according to the current topic information.
- Step 2: If the user has done this question, the answer data will be updated according to the user's answer.
- Step 3: The update buttons. if the title is the first question, you cannot use the previous button, if the title is the last question, you cannot use the next button.

5.6. Implementation of Submitting Answer's Function

When the student press “the submit test paper” button or the timer returns to zero, the ExamService (the program's core business) will execute the commit command to calculate the candidate's scores of each type of question and save the result into the score table of the database. Next, let's take the judgment topic as example. The execution process of the whole program is as follows:

First of all, the score is calculated by the score settlement function of commit command. The steps are as follows.

- Step 1: The result of judging questions will be calculated by cyclic calculation.
- Step 2: The corresponding bonus operation will be performed according to the consistence of the user answer and the correct answer.
- Then, it submits the results through the results submission function in the commit command, and the steps are as follows.
- Step 1: Calculate the total score of the examination through adding the scores of each type of question.
- Step 2: Read the information in the Properties profile through the Config class and connect it to the database.
- Step 3: By using SQL insert command, the test results are stored in the database and then disconnected from the database.
- Step 4: Set the test status to complete.

5.7. Implementation of Server Response

The data transmission between the student side and the server side uses a simple Socket transmission protocol, in which the reflection mechanism is used to execute the corresponding command. The whole reposing process is shown in Figure. 11.

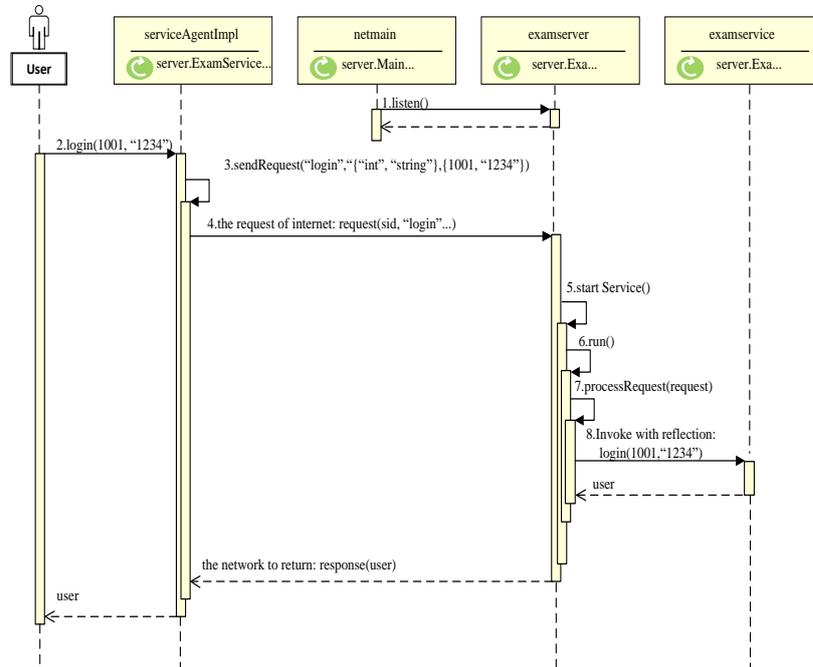


Fig. 11. The whole response process of the server

Before the exam, the Socket connection is established by starting the server with the netmain class (server-side main program). Then the listen thread is executed to start listening requirements. When the user passes the login (1001, '1234') command to the serviceAgentImpl (the network function implementation class of core business), the class encapsulates the login request as a request class, which contains an empty sid, the command name (login) of the command, the parameter type (the first parameter type is int, the second parameter type is String), and the parameter value (the first parameter value is 1001, the second parameter value is '1234'). Then the request is sent to the Server (the server side), which gives a sid. Then, take the sid as the key, an ExamServiceImpl (the business core function implementation class) as the Value to join a HashMap. After, the login (1001, '1234') method in the realization class of the core business function is called through the reflection mechanism, and a user parameter is returned to be encapsulated into a response return, which is parsed into a user parameter.

At this point, the transmission of the command is ended. When the user submits the request command again, the corresponding business core function implementation class can be found from the HashMap according to SID, and the corresponding command can be executed through reflection mechanism.

6. Conclusion

With the promotion of online education, online evaluation system has attracted more and more attention. This paper introduces the function design, process analysis and implementation of online evaluation system, especially the process of file transmission based on socket protocol. Practice shows that the system has good stability and security and it can meet the needs of online evaluation. In the future research, the personalized test of online test can be further improved. In the case of collecting more data, the difficulty of the test can be divided more reasonably. In general, our study provides a reference for the design of online evaluation system.

Acknowledgment. This work is supported by the project of Shanghai Philosophy and Social Sciences Plan (No.2016BGL004), National Natural Science Foundation of China (No.71971066) and Project of key education and scientific research in Jingan District of Shanghai (zs202102).

References

1. Alkhafaji, S., Sriram, B.: Instructor's Performance: A Proposed Model for Online Evaluation. *International Journal of Information Engineering and Electronic Business* 5(4), 34-40 (2013)
2. Asare, S., Daniel, B.K.: Factors Influencing Response Rates in Online Student Evaluation Systems: A Systematic Review Approach. *Journal of Interactive Learning Research* 29(2), 133-143 (2018)
3. Castillo, I., Pascual, V.: The WebSocket Protocol as a Transport for the Session Initiation Protocol (SIP). *Journal of Biosciences* 33(3), 309-311 (2013)
4. Chen, J.: Analysis and Study of an Online Assessment System for the Data Structures Course. *Intelligent Computer and Applications* 10(06), 264-267 (2020)
5. Cormier, S.M., Zheng, L., Hill, R.A., Nova, R.M., Flaherty, C.M.: A flow-chart for developing water quality criteria from two field-based methods. *The Science of the Total Environment* 633(15), 1647-1656 (2018)
6. José, L.P.L., Augusto, C.P., Rocío, A.M.: Analysis of the Academic Management and Assessment of External Placements from the University of Valladolid. *Procedia Social and Behavioral Sciences* 139(2014), 487-495 (2014)
7. Kang, H.Y., Fan, X.Z., Tang, S.P.: Research and Design of Online Test-evaluating System Based on J2EE. *Computer Engineering* 13, 169-171 (2004)
8. Kurt, H.S., Doan, Z.: Pre-Service Science Teachers' Skills to Express The Algorithms Used in Solving Physics Problems with Flowcharts (An Example From Turkey). *Jurnal Pendidikan Fisika Indonesia* 16(1), 24-33 (2020)
9. Li, Y., Yang, G.B., Ding X.L., Zhu, Y.P.: Early DIRECT Mode Decision for MVC Using MB Mode Homogeneity and RD Cost Correlation. *IEEE Transactions on Broadcasting* 62(3), 700-708 (2016)
10. Lorentz, J., Sorana-Daniela, B.: Auto-calibrated Online Evaluation: Database Design and Implementation. *Leonardo Electronic Journal of Practices and Technologies* 5(9), 201-204 (2006)
11. Mo, Z., Zhang, A., Yang, Z.: A new parallel algorithm for vertex priorities of data flow acyclic digraphs. *Journal of Supercomputing* 68(1), 49-64 (2014)

12. Moghadam, N., Li, H.: A New Wireless Multicast Queuing Design Using Network Coding and Data-Flow Model. *IEEE Communications Letters* 20(8), 1603-1606 (2016)
13. Monika., Upadhyaya, S.: Secure Communication Using DNA Cryptography with Secure Socket Layer (SSL) Protocol in Wireless Sensor Networks. *Procedia Computer Science* 70, 808-813 (2015)
14. Moskal, A.C.M., Stein, S.J., Golding, C.: Can you increase teacher engagement with evaluation simply by improving the evaluation system?. *Assessment and Evaluation in Higher Education* 41(2), 286-300 (2016)
15. Murugan, P.V., Queen, V.M.: MOOCs as a Digital Learning Platform. *International Journal of Multidisciplinary Research Review* 3(1), 28-37 (2020)
16. Rienties, Bart.: Understanding academics' resistance towards (online) student evaluation. *Assessment and Evaluation in Higher Education* 39(8), 987-1001 (2014)
17. Tucker, B., Jones, S., Straker, L.: Online student evaluation improves Course Experience Questionnaire results in a physiotherapy program. *Higher Education Research and Development* 27(3), 281-296 (2008)
18. Wang, D.M., Ding, L., Li, G.J.: Research on OA System Development Platform Architecture of MVC Mode. *Applied Mechanics and Materials* 421, 690-693 (2013)
19. Wang, M., Yan, Z., Wang, X.: Design and Implementation of Home Heating Intelligent Management Application on iOS Mobile Platform. *Guide of Science and Education* 17(10), 1-12 (2015)
20. Wang, S.M.: On-line Examination System Based on Browser/Server Mode. *Computer Technology and Development* 1, 59-60 (2014)
21. Wang, Z.X.: Summary of the Development of Internet Online Education. *Creative Education Studies* 3(4), 164-167 (2015)
22. Wojciechowski, J., Sakowicz, B., Dura, K., Napieralski, A.: MVC model, struts framework and file upload issues in web applications based on J2EE platform. *Modern Problems of Radio Engineering, IEEE International Conference Telecommunications and Computer Science* 342-345, (2004)
23. Wu, H.L., Cheng, Y.H.: Design of the Logistics Management System Based on J2EE and MVC. *Advanced Materials Research* 765-767, 1419-1422 (2013)
24. Yviquel, H., Boutellier, J., Raulet, M., Casseau, E.: Automated design of networks of Transport-Triggered Architecture processors using Dynamic Dataflow Programs. *Signal Processing Image Communication* 28(10), 1295-1302 (2012)
25. Zhang, J., Pu, X., Zhang, Z.: Design and Implement of Teaching Resources Management Network Platform Based on MVC. *Applied Mechanics and Materials* 631-632(2), 999-1002 (2014)
26. Zhang, L.Q., Li, Y.: Design and Implementation of College Program Online Evaluation System Based on B/S. *Communication and Information Technology* 4, 33-36 (2018)
27. Zhang, W.S., Chen, H.: The Research and Application of Modular Mobile Phone Web Front-end Based on MVCS Mode. *International Journal of Future Generation Communication and Networking* 8(5), 97-106 (2015)
28. Zhang, Y.F., Ke, C.Y.: Applied Technology in an Interactive Design for a Web-Based Language Teaching System. *Advanced Materials Research* 886, 621-624 (2014)
29. Zhong, Y.Z., Gui, Q.: The Design and Implementation of the ACM Competition Online Evaluation System. *Wireless Internet Technology* 17(18), 42-44 (2020)

Peng Jiang is an associate professor at Jingan Branch Campus, Shanghai Open University, China. His current research interests include Educational technology and Management Information System. Contact him at jzhpmail@163.com.

Kexin Yan is the corresponding author of this paper, she is a master at the Management School, Shanghai University of International Business and Economics, China. Her current research interest is the Data Science and Management Information System. Contact her at ykx980219@163.com.

Haijian Chen is a professor at the Shanghai Academic Credit Transfer and Accumulation Bank for Lifelong Education, China. He received his Ph.D. in Management Science and Engineering from Shanghai University of Finance and Economics, China in 2015. His current research interests include Educational technology and cloud computing. Contact him at xochj@sou.edu.cn.

Han Sun is a lecture at management school, Fudan University, China. He received his Ph.D. in Management Science and Engineering from Tongji University, China in 2002. His current research interests include Management Information System and big data analysis. Contact him at sunhai@fudan.edu.cn.

Received: February 01, 2021; Accepted: July 30, 2021.