

# DESIGN AND IMPLEMENTATION OF HETEROGENEOUS DATABASE INTEGRATION BASED ON NOSQL

Shuo Wang\*, Linfeng He

School of Computer Science, Huazhong Agricultural University, Wuhan, Hubei, 430000, China

## Abstract

With the arrival of the era of big data, the Internet generates a large amount of data every day, and the type of data gradually inclines from structured data to unstructured data, which brings great challenges to traditional relational databases. At this time, the emergence of non-relational database provides a good solution for dealing with unstructured data in the era of web 2.0. The object of this study is the integration of non-relational database (NoSQL) with traditional database functions. On this basis, a comprehensive student assessment management system based on MongoDB database is designed and implemented.

**Keywords:** Unstructured data; MongoDB database; Student Comprehensive Measurement Management System.

## INTRODUCTION

With the arrival of the era of big data, people have to produce a large amount of data in daily life, including a large number of audios, video and other unstructured data. How to deal with and store these huge data is a key issue. For massive data, the query efficiency of relational database is very low, and the scalability and availability of database are low, and the performance of high concurrent reading and writing is low, which leads to the low efficiency of database mutual access between different systems. Moreover, for many systems that need to provide uninterrupted services, the database system is implemented. Upgrades or extensions often require downtime maintenance and data migration.

NoSQL database is a non-relational database. With the arrival of the era of big data, the defect of relational database becomes more and more obvious, and non-relational database emerges as the times require. NoSQL database supports the storage and processing of unstructured data, and has high scalability, high data compatibility and meets the requirements of large data processing.

At present, non-relational databases are not document databases, column databases, Graph databases, and Key-Value databases. Among them, document is the basic storage unit in document database, the data structure is not strict, the table structure is variable, and it does not need to define the table structure in advance like relational database, so it is suitable for web application development; column storage database compresses and stores the same column of data together, which has fast searching speed and extensibility. It is suitable for distributed development. Graph database uses graph structure correlation algorithm, and it is suitable for graph relation mining and other projects. Key and Value key pairs are used to store information, and hash tables are usually used to achieve fast search speed. Each distributed database has its own special application

scenario, which is different from the traditional relational database and more suitable for storage and processing of large data and unstructured data.

MongoDB is a product between relational database and non-relational database, which has the most abundant functions and resembles relational database. It belongs to document database. It supports a very loose data structure, similar to json's bson format, so it can store more complex data types. The biggest feature of MongoDB is that the query language it supports is very powerful. Its grammar is somewhat similar to object-oriented query language. It can almost achieve most functions similar to single-table query in relational database, and it also supports the indexing of data.

Comprehensive evaluation refers to the establishment of a series of quantitative indicators and Implementation Rules for college students 'morality, intelligence and physique. It is an important basis and reference for college students' scholarship, evaluation of three good students and employment of graduates, and it is closely related to every college student.

In the past, the process of comprehensive assessment is very complicated, consumes a lot of manpower, and needs to collect and process a large number of student data. The data types and structures are diverse, and the scoring standards in different aspects are also diverse. The integrated measurement management system developed in this project automatically realizes the evaluation of integrated measurement, reduces the repetitive workload, improves efficiency, and facilitates users to query.

## 1. MATERIALS AND METHODS

### Achieving goals

Using MongoDB database, a student comprehensive assessment management system is realized. At the same time, non-relational database is used instead of relational

database to store students' multi-dimensional information. At the same time, the student comprehensive assessment management system is set up as a system platform for counselors, student management and inquiry. Its corresponding users are counselors and student affairs workers. Authors, ordinary students, and provide queries, modifications and other corresponding operating interfaces.

#### **data description**

Data file information: students' transcripts and records of rewards and punishments for activities, etc. Dynamic data refers to data that changes with time in system applications. The preparation of dynamic data is directly related to the time of system switching. Dynamic data is often changing, which directly reflects the transaction process. Input data: user account, password; rewards and penalties, leave, self-evaluation and mutual scores. Output data: leave, rewards and punishments, student achievement calculation, student total score ranking.

Database: MongoDB 3.4

Its advantages lie in: It can run on all mainstream platforms (including Windows and Linux, macOS). Collection-oriented storage, easy to store object-type data. The model is free, supports dynamic queries, and uses efficient binary data storage, including large objects.

#### **Operating environment**

User Interface: Using graphical interface

Hardware Interface:

Processor requirements: Pentium VI or above

Running environment: WebStorm

Memory requirements: 512MB or more

Software Interface

Operating System Platform: Microsoft Windows 10

Database System Platform: MongoDB 3.4

Development Language: JavaScript Html CSS

Development tools: WebStorm

Fault Handling: Users will automatically jump back to the login if their account password is incorrect when they log in to the system.

#### **Design instructions**

There are three types of interfaces: students, student cadres and counselors. The student interface should include: personal profile, self-evaluation and mutual evaluation, award items, minus points, feedback interface. The interface of student cadres should include performance management, reward and punishment management and feedback management. The counselor interface should include: examination and approval of grades, general situation of grades, examination and approval of leave, examination and approval of students' rewards and punishments.

Main interfaces:

Singnup: User registration interface, the system has set the initial login account and password for students.

Singin: User Login Interface

Assess: User self-evaluation and mutual evaluation

Carderewards: Reward and Punishment Management Interface for Student Cadres

Cardegrades: Examination and Approval Interface for Student Cadres' Achievements

Student: Student's Main Interface

Teacher: Counselor's Home Interface

Feedback: Feedback management interface

General: Counselors view the overall ranking of the entire grade

Grades: Management of Student Achievement Management

Punishment: Students' minus points

Situation: Students' Personal Profile Interface

Checkleave: Counselor's leave approval interface

Leave: Student leave interface

Main methods: Various Logics of Interface and Realization of Interface Jump. The jump of user interface is judged by logic in JS and then realized.

Major modules files:

Activity:

Assess: Connect to Student database

Awards: Connect to rewards database

Leave: Connect to Leave database

Grades: Connect to Student database

General: Connect to Student database

Feedback: Connect to the feedback database

Schema file:

Award: Defines the attributes of student performance in a database

Feedback: Defines the attributes of feedback in the database

Managerewards: Defines the attributes of rewards and penalties in the database

Student: Defines the attributes of students in Databases

Main control documents:

Cardecontrolles: This control file mainly realizes the logical control of student cadres.

Feedback: Feedback submitted by students is handled on the feedback interface submitted to student cadres.

Rewards: Logical Control of Student Cadres Filling in Rewards and Punishments

Grades: Submit the data needed to be displayed in Student database to the achievement management interface of student cadres

Controllers: This control file mainly realizes the logical control of students and counselors.

## **2. DISCUSS**

The MongoDB database is not connected: the Mongo database is not started or needs to be restarted, or the Mongoose module is not imported, or the local database is not configured;

Excel table extraction is problematic: the js-XLSX plug-in supports. xlsx format tables, when the table is. xls, it cannot start the function correctly, and the contents of the table can be moved to a new. xlsx table for storage;

Some webpages display results that are not in line with expectations: check whether the style label of EJS file is correctly modified, if there is no problem, check the referenced CSS file, check and modify after checking. The program cannot start correctly: check whether there is an error or not, and modify it according to the prompt. The main problem is the error of code and the occupation of port.

Some Mongo statements report errors: check if there are any grammatical problems, and then clarify the business logic. Whether the data returned is a single document or multiple documents, the general problem that null or length does not define is that.

How to solve the problem: If there is a problem on the page, you can look at the source code of the page to determine the location of the problem. Some keyword errors can be searched globally to see the location of the call. Intermediate variables can be displayed in debug to identify the problem location more quickly.

Interface jump forwarding error: First, check whether tag a of EJS file is logical when jumping, check whether the function called belongs to the exclusive control file, and then go to router. JS to see if the route is correct.

The business of multiple identity users is confused: to view the code where confusion may occur, or to separate the controllers of users with different privileges.

MongoDB allows scripts to be executed on the server side. A function can be written in JavaScript, executed directly on the server side, or the definition of the function can be stored on the server side. Next time, it can be called directly. MongoDB supports a variety of programming languages: RUBY, PYTHON, JAVA, C++, PHP, C # and other languages. Express as an efficient framework also provides us with convenience for development. As a mature non-relational database, MongoDB has a high efficiency in processing large data, but the advanced features used in this project are not many, such as fragmentation mechanism, disaster recovery mechanism and so on. On the one hand, it is limited by our technical level, on the other hand, it is limited by the resources we can use. It is hoped that the project can be improved and perfected later.

### 3. CONCLUSIONS

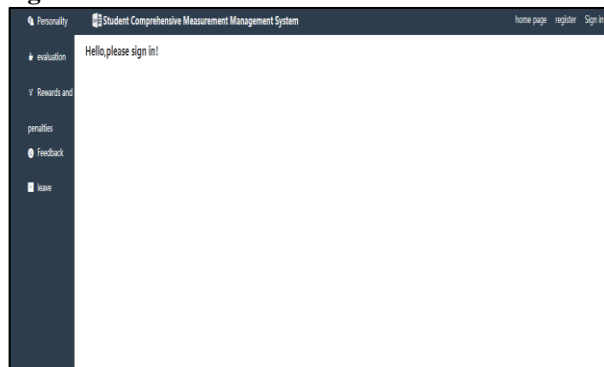
The student comprehensive assessment management system is mainly for University students. Comprehensive evaluation refers to the establishment of a series of quantitative indicators and Implementation Rules for college students 'morality, intelligence and physique. It is an important basis and reference for college students' scholarship, evaluation of three good students and employment of graduates, and it is closely related to every college student. The background database of the system contains the basic information of students 'names, classes, academic year results, records of rewards and punishments, records of practical and innovative abilities, etc., and automatically settles corresponding scores in the background, gives rankings, facilitates counselors, student cadres and students to inquire.

This system requires the user to output the username and password to log in. User objects are divided into three roles: ordinary students, student cadres and counselors. Different roles correspond to different functions of the application system. The initial interface of the system is shown in **Figure 1**.

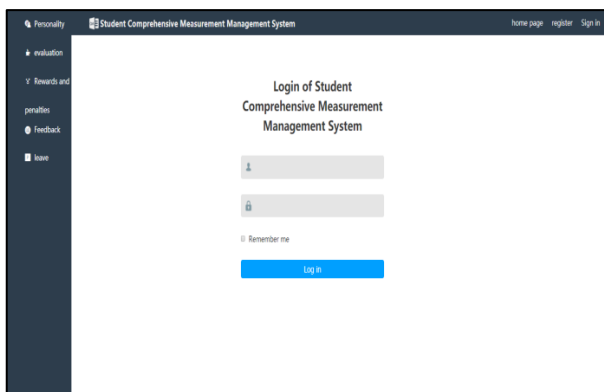
For the sake of privilege and privacy, account number and password are preset. Student users can login according to their names and initial passwords (8.1, can be changed) to operate student-related affairs; student

cadres correspond to the user name "xueshengganbu", the initial password is 12345678 (can be modified), after login, they can operate student cadres 'related affairs; and counselors correspond to the user name "fudaoyuan". The initial password is 12345678 (which can be modified). After login, the counselor can operate related affairs. The system login interface is shown in **Figure 2**.

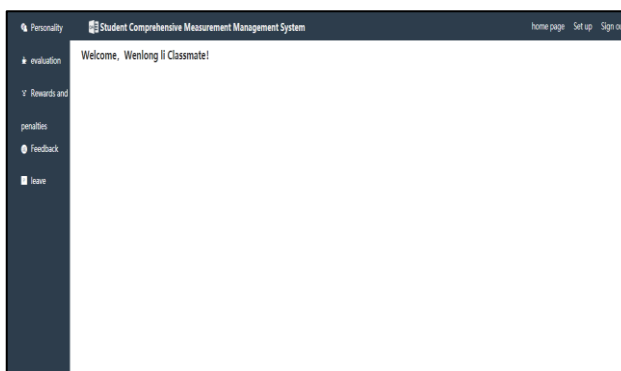
After entering the correct user name and password, enter the system, the system operation interface as shown in **Figure 3**.



**Fig. 1 Initial Interface of Student Comprehensive Measurement Management System**



**Fig. 2 Login Interface of Student Comprehensive Measurement Management System**

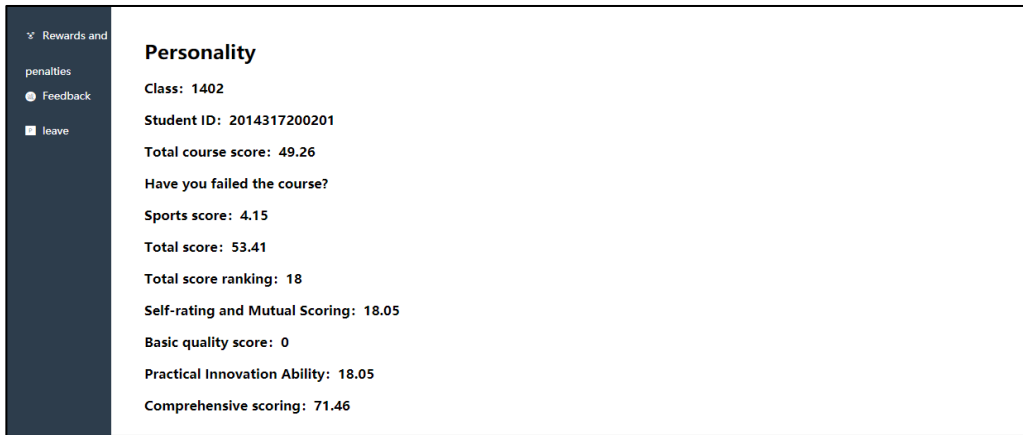


**Fig. 3 System Operating Interface of Student Comprehensive Measurement Management System**

Student users can view their personal profiles and rewards and punishments for the academic year through the system. Students can feedback the query results, and students can also ask for leave through the system. In this module, the logged-in students can check their names, classes, school numbers and other basic information. At the same time, they can check their total

course scores, whether they have failed the course, the sports scores of the school year, the total score of the school year (the sum of the total course scores of the school year and the sports scores of the school year), and self-evaluation each other. Scores (total score of 20 points), basic quality score (sum of reward score of participating activities and punishment score of daily affairs), practical innovation ability score (sum of basic quality score of self-evaluation and mutual evaluation),

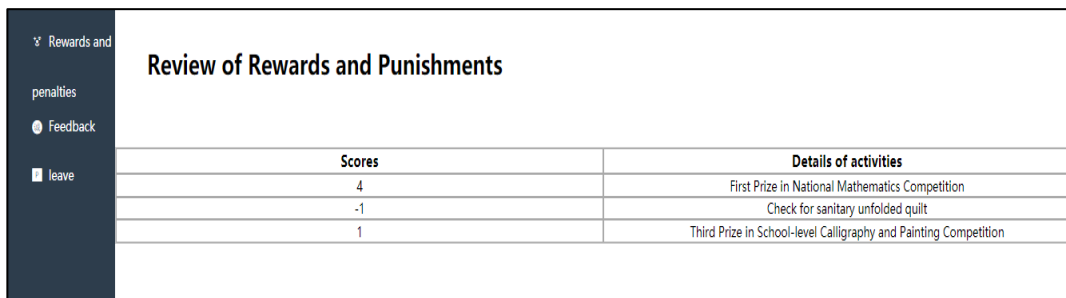
comprehensive evaluation score (total score of comprehensive evaluation in this academic year, total score of achievement and practical ability creation) New score sum), and the corresponding ranking of the total score. It directly and omni-directionally presents the overall performance of students in the academic year, and reduces the workload of manual calculation. Personal profile function module is shown in **Figure 4**.



**Fig. 4 Personal Profile Function Module Interface of Student Comprehensive Measurement Management System**

In this module, the logged-in students can view the items and scores of their participation in activities reward and daily affairs punishment in this academic year as part of

the comprehensive evaluation reference. The reward and punishment viewing module is shown in **Figure 5**.



**Fig. 5 The interface of reward and punishment checking function module of student comprehensive assessment management system**

In this module, logged-in students can make error feedback based on their own personal profile and the results of rewards and punishments, and the feedback results will be handled by student cadres. Students can click on the "Add Feedback Information"

option at the top right to submit feedback messages, including feedback questions and feedback content. You can also move the mouse to the right of the screen, with the options of "modify" and "delete" to operate. Add feedback interface as shown in **Figure 6**.



**Fig. 6 Adding Feedback Information Interface to Feedback Function Module of Student Comprehensive Measurement Management System**

In this module, logged-in students can ask for leave from counselors online through this system, and counselors can examine and approve students' leave in the background. Similarly, students can modify and delete the leave message. At the same time, students can click

the "Fill in the leave slip" option at the top right to submit the leave message. The content includes the leave applicant, class, leave type, application time, leave start and stop time and reasons. Fill in the leave bar interface as shown in **Figure 7**.

**Fig. 7 The Interface of Filling in Leave Bar in Leave Function Module of Student Comprehensive Measurement Management System**

Student cadres mainly assist counselors in related work, mainly in the management of rewards and punishments, filling in the corresponding rewards and punishments for students; there is also an overview of students' achievements, which is convenient for viewing when dealing with feedback; finally, the feedback information of ordinary students is checked and processed in a unified way. In this module, the logged-in student cadres will fill in the corresponding rewards and punishments for students, check the specific rewards and punishments items, and delete the specific rewards and punishments items. Student cadres can click on the "Adding Student Rewards and Punishments" option at the top right to submit information about students' rewards and punishments, including their names, school numbers, details of activities and corresponding scores. Multiple students can be added to the same activity. You can also move the mouse to the right of the screen, with the options of "View" and "Delete" to operate. In this module, the logged-in student cadres can view the overall performance of the grade students through this system, which is convenient for students to check when they give feedback.

## REFERENCES

- [1] Yang Dong, Sheffy, Yang Xiaogang, He Zunwen, etc. Research and implementation of distributed database technology suddenly. [Journal Papers] - Electronic Science and Technology 2015 (1)
- [2] Xu Aiping, Song Xianming, Xu Wuping, et al. Research and implementation of distributed heterogeneous database integration system. Computer Engineering and Science, 2015, 37 (10): 1909-1916
- [3] Yu Fan, Wang Zhenze, Wang Zhenhui, etc. Design of XML-based heterogeneous database integration middleware. Computer Applied Research, 2007, 24 (9): 185-188
- [4] Yang Qiaoguang. Design and Implementation of Intelligent Campus System Based on Heterogeneous Database. South China University of Technology, 2016
- [5] E Puget Hawkins, P Membrey. The Definitive Guide to MongoDB: The NoSQL Database for Cloud and Desktop Computing. 《Springer Ebooks》, 2010
- [6] R Lawrence. Integration and Virtualization of Relational SQL and NoSQL Systems Including MySQL and MongoDB. International Conference on Computational Science, 2014, 1 (3) :285-29
- [7] Pereira D A , Wagner O D M , Edison P D F . NoSQL real-time database performance comparison[J]. International Journal of Parallel, Emergent and Distributed Systems, 2017:1-13.
- [8] Saur K , Dumitras T , Hicks M . Evolving NoSQL Databases without Downtime[C]// 2016 IEEE International Conference on Software Maintenance and Evolution (ICSME). IEEE, 2016.
- [9] Mahgoub A , Chaterji S , Wood P , et al. Rafiki: a middleware for parameter tuning of NoSQL datastores for dynamic metagenomics workloads[C]// the 18th ACM/IFIP/USENIX Middleware Conference. ACM, 2017.
- [10] Reniers V , Rafique A , Van Landuyt D , et al. Object-NoSQL Database Mappers: a benchmark study on the performance overhead[J]. Journal of Internet Services and Applications, 2017, 8(1):1.
- [11] Jeong H , Choi J , Choi C , et al. A Design of Web Log Integration Framework Using NoSQL[C]// Information and Communication Technology - EurAsia Conference. 2014.