Design and Implementation of a College Student Sports Performance Management System Under Web Technology

Yunlong Ma[1,a*], Chengxue Zhang[1,b], Hui Yang[2,c]

[1]Weifang Engineering Vocational College, Qingzhou, Shandong Province, 262500, China
[2]No.2 Middle School of Qingzhou Shandong, Qingzhou, Shandong Province, 262500, China

[a*]1278407287@qq.com
[b]417657274@qq.com
[c]1607013759@qq.com

Abstract. This system follows user-centric and business process modeling principles, comprising four major functional modules: performance management, physical testing management, statistical analysis management, and system configuration. The performance management module is responsible for student performance data entry, modification, verification, and publication; the physical testing management module handles data collection and physical assessment; the statistical analysis management module provides multidimensional data statistics and analysis; the system configuration module is responsible for maintaining basic data and permissions. These modules are integrated through a common service component layer, including identity authentication, data access, rule engine, and workflow engine. The statistical analysis module supports multidimensional data aggregation and various report formats, making it convenient for management and teaching assessment. The modular and service-oriented design of the system achieves high cohesion and loose coupling of functions, facilitating future expansion and maintenance.

Keywords: College sports management system; service-oriented architecture; performance management; statistical analysis; system implementation.

1 Introduction

The task of managing sports performance in colleges is both burdensome and inefficient. Many teachers need to manually calculate and record various types of assessment results for students, which not only consumes a lot of time but also is prone to errors [1]. At the same time, the lack of statistical analysis of student physical data makes it difficult to diagnose and improve physical education and management. Therefore, the development of an efficient student sports performance management system is urgently needed. This system needs to be designed from the perspectives of business process reengineering and technological innovation, meeting the requirements of users for

M. Yu et al. (eds.), *Proceedings of the 2024 5th International Conference on Big Data and Informatization Education (ICBDIE 2024)*, Advances in Intelligent Systems Research 182,
https://doi.org/10.2991/978-94-6463-417-4_42

efficiency and convenience while ensuring data security and compliance. By integrating advanced network, software, and database technologies and adopting a flexible system architecture, it can efficiently support sports management operations and provide high-quality information services [2]. The goal of this paper is to design and implement a web-based system for college sports performance management. Through specific development practices, the selected technological solution will be validated, and experience will be accumulated for future promotion and application.

2 Requirements Analysis of the Sports Performance Management System

Sports performance management is a crucial aspect for universities. On the one hand, sports performance accounts for 5% of the overall academic evaluation, directly impacting the eligibility for scholarship consideration[3]. An analysis of scholarship assessment data from the past three years reveals that 15% of scholarships were disqualified due to insufficient sports performance. On the other hand, based on the physical fitness reports of 2,000 students from the 2019 cohort, nearly 30% of students fail to meet the national standards for physical fitness. Among these, the most severe cases involve poor cardiovascular fitness, constituting 65% of those deemed unfit. Currently, university sports performance management relies heavily on paper records and manual calculations, with approximately 20,000 student scores requiring manual processing each semester. This approach is not only inefficient but also prone to high error rates. Investigations indicate that the error rate in sports performance grades across various departments is around 3%. Therefore, the development of an efficient student sports performance management system is of paramount importance. This system primarily targets the sports education management departments and students, offering functionalities such as the collection of various physical fitness test data, performance assessment, statistical analysis, and result inquiries[4]. This aims to enhance the current inefficient management practices[5].

3 Overall Design of the Sports Performance Management System

3.1 Design and Implementation of the Network Architecture

The sports performance management system adopts a Browser/Server model in its network architecture design, utilizing a typical three-tier architecture consisting of the presentation layer, business logic layer, and data access layer, as illustrated in Figure 1[6]. To support future performance optimization, the network design includes a cache server and upgraded network equipment. The presentation layer is deployed in the client's browser, responsible for user interaction and interface display. The business logic layer is deployed in the Tomcat application server, handling the system's core business processes. The data access layer utilizes the MySQL database server, managing data persistence and access control. The network topology design follows a star-shaped structure, with the school's data center serving as the network core. The sports performance management server, database server, and cache server are situated within this core[7]. The data center's network equipment is upgraded to 40Gb bandwidth, and redundant fiber optic links connect the sports performance management and cache servers. Each college and sports department is connected to the data center via the campus network, and students and teachers access the system server through their respective local area networks. Simultaneously, the system supports multi-user access in computer classrooms. This network architecture design aligns with the characteristics of a Browser/Server system, achieving centralized storage of sports performance data, centralized deployment of system modules, supporting distributed multi-user access, and enhancing system security and efficiency[8].

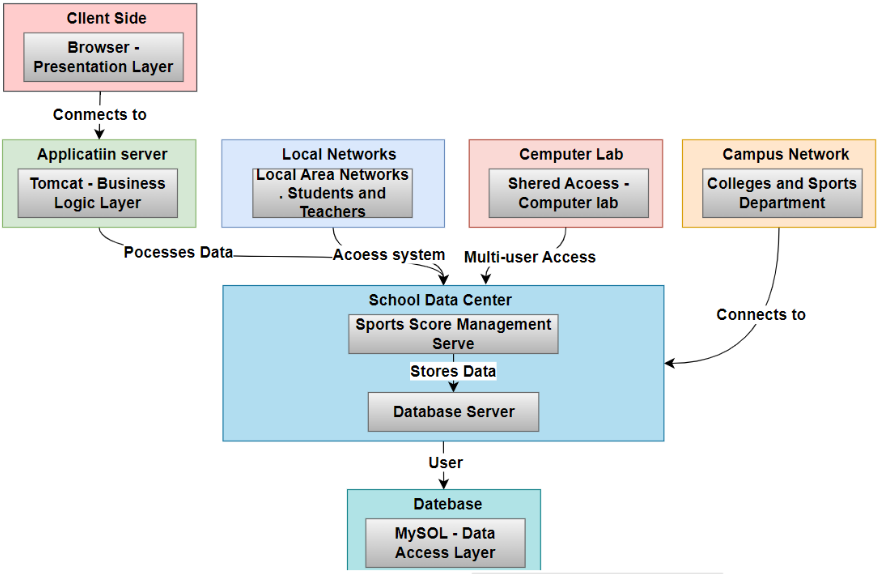

Cllent Side
Browser - Presentation Layer
Connects to
Applicatiin server
Tomcat - Business Logic Layer
Local Networks
Local Area Networks . Students and Teachers
Computer Lab
Shered Acoess - Computer lab
Campus Network
Colleges and Sports Department
Pocesses Data
Acoess system
Multi-user Access
Connects to
School Data Center
Sports Score Management Serve
Stores Data
Database Server
User
Datebase
MySOL - Data Access Layer

Fig. 1. Design and Implementation of the Network Architecture

3.2 Integrated Design of Functional Modules

This system follows the principles of user demand orientation and business process modeling and includes five major functional modules: Grade Management, Physical Test Management, Statistical Analysis Management, System Configuration, and Security Management [9]. The Grade Management module is responsible for entering, modifying, reviewing, and publishing student grades. The Physical Test Management module handles data collection and physical fitness assessment. The Statistical Analysis Management module provides multidimensional data statistics and analysis through optimized query algorithms. The System Configuration module is responsible for maintaining basic data and permissions. The Security Management module controls the security of sensitive information such as student grades and physical test data, preventing data leaks and other security incidents. The Public Service Component Layer provides performance monitoring, high availability support, as well as identity authentication, data access, rule engine, and workflow engine services. The system's modular and service-oriented design achieves high cohesion and loose coupling of functions, making it convenient for future expansion and maintenance.

```
# Define the Score Management Module
class ScoreManagementModule:
    def __init__(self):
        # Initialize the Score Management Module
    def input_score(self, student_id, subject, score):
        # Enter student scores
        # In the actual code, perform database operations to store the student's score in the database
        pass
# Main program
if __name__ == "__main__":
    # Create an instance of the Score Management Module
    score_module = ScoreManagementModule()
    # Enter student scores
    student_id = "12345"
    subject = "Math"
    score = 95
    score_module.input_score(student_id, subject, score)
```

3.3 Database Architecture and Design Strategy

The sports performance management system has selected MySQL as the database management system and utilized a three-tier architecture for table structure design [10]. The database comprises three fundamental tables: the basic information table, the physical fitness test score table, and the operation log table. To optimize query performance, a composite index has been added to the physical fitness test score table. The basic information table stores static foundational data such as student and course information, stored in the form of codes and text descriptions. The physical fitness test score table stores various test score data, employing array storage to support multiple tests for the same student. The operation log table records crucial operational logs for system functioning and serves as the basis for system security auditing. Foreign key relationships are established between tables through columns like student ID and course ID. Additionally, the database utilizes mechanisms such as stored procedures, triggers, and views to further encapsulate access logic, enhancing security and scalability. Moreover, the database has enabled parallel query functionality, supporting simultaneous execution of multiple query tasks and significantly reducing response times. For sensitive information such as student grades and fitness test data, the system employs security control measures including encrypted storage, access control, and operation logging. Grade data in the physical fitness test score table undergoes AES symmetric encryption during storage, and decryption keys are securely managed. The database implements authentication and authorization mechanisms, restricting operators to access only authorized student data. Detailed logs are generated for each data access operation, facilitating subsequent audits. These security measures prevent unauthorized access or leakage of data. Specific design criteria for the database include data accuracy ≥99%, individual table data volume ≤500,000 rows, and response time ≤0.5 seconds, among other specifications.

$$A = \frac{C}{T} \times 100\% \tag{1}$$

Where A represents data accuracy, C represents the number of correct data entries, and T represents the total number of data entries. In the actual implementation, this system adopts a code-first strategy, using the Hibernate framework to map class-object relationships to the database table structure. This object-oriented model encapsulates the data access layer, reducing coding work and facilitating centralized development of business logic. It also allows for easier database replacement and provides better scalability.

4 Implementation and Development of the Sports Performance Management System

4.1 Development Environment and Implementation of the Authorization Module

The development of this system follows the J2EE three-tier architecture, with Apache Tomcat as the web server and MySQL as the database. During development, version control is managed using Gitee, dependencies and builds are handled through Maven, and automated testing is conducted with JUnit. The core of the system is an authorization and authentication module, providing password login and single sign-out services, as well as integrating with the campus unified identity authentication system for single sign-on. This module utilizes the Spring Security framework and integrates with the LDAP server for campus card authentication, customizing a UserDetailsService to load user information. The system is designed with three role-based permissions: student, teacher, and administrator, each associated with different functional access rights. This fine-grained permission control model enhances the system's security and ensures effective enforcement of functional and data access permissions for different user types. Through the authorization module, the system can integrate with the campus unified identity authentication system, providing secure and reliable user identification and access control support for the business functional modules, as shown in Figure 2.

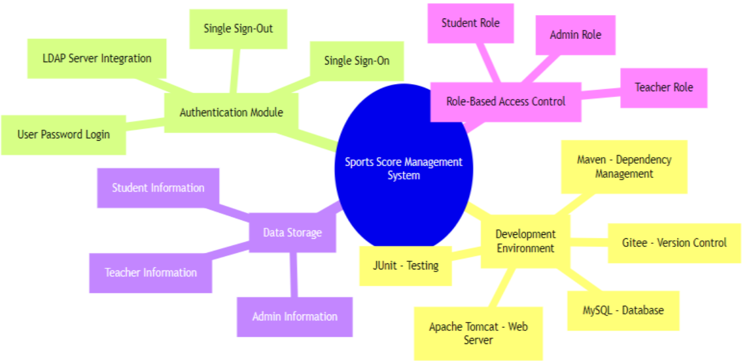


Fig. 2. Development Environment and Authorization Module

4.2 Development of the Student Performance Management Module

The core of the system's development is the student performance management module, designed for teacher users, providing functions for entering, modifying, reviewing, and publishing grades. This module includes the entry sub-module, review sub-module, and query sub-module. The entry sub-module allows for batch import of grades based on an Excel template, using Apache POI to parse Excel documents and store data in the database in bulk, improving work efficiency. The review sub-module conducts grade reviews through a predefined workflow, and only grades that pass the review can be published. The query sub-module provides multidimensional grade query capabilities, allowing teachers to view the entire class's grades. The GUI interface is implemented using the Vue framework, and data is stored in the grade database table, with data exchange handled through the Spring MVC DAO layer. This grade management module provides crucial support for school teaching management.

4.3 Implementation of the Statistical Analysis Module

The statistical analysis module is primarily implemented based on the Spring container, using Hibernate for database operations, and ECharts for generating statistical charts. The module can aggregate data based on different dimensions such as students, classes, and colleges, enabling the generation of visual reports such as grade distribution, trends, and radar charts. The reports are designed hierarchically and support flexible filtering and multiple format exports. The module optimizes query performance using mechanisms like indexing and caching, ensuring that response times for analysis are kept within 5 seconds, even with large datasets. It interfaces with the grade management module to directly retrieve grade data stored in the MySQL database. The module provides statistical services to the outside world in the form of RESTful APIs, encoding responses in JSON format, making it easy for other systems to call and for secondary development.

```
// Hibernate Configuration
@Configuration
```

```
public class HibernateConfig {
// Configure SessionFactory, etc.
}
// Example of a RESTful API
@RestController
@RequestMapping("/api/stats")
public class StatisticsController {
```

5 System Testing and Performance Evaluation

5.1 Testing Methodology and Test Case Design

The testing of this system adopts a comprehensive testing approach that combines black-box testing and white-box testing. Test cases are designed based on the business logic of functional modules. The test case design follows the principles of synchronizing with development and partitioning functionality, covering main process scenarios and exceptional scenarios for each module. A total of 20 test cases are designed for the data entry module, 15 for the approval module, and 12 for the query module. Test case content includes: case name, testing purpose, input data, execution steps, and expected results. Simultaneously, considering the quality requirements for usability, a usability testing plan is designed by combining questionnaire surveys and user observation. The survey focuses on aspects such as the humanization of the system interface and the convenience of operation processes. Considering security requirements, a security penetration testing plan based on the OWASP methodology is designed, covering common security threat models such as injection vulnerabilities, cross-site scripting, and denial-of-service. To assess system performance metrics under high-concurrency scenarios, stress testing cases are designed. Simulating a scenario where 1000 users access the system simultaneously, the goal is to evaluate whether the peak response time meets the design target of less than 2 seconds. The testing environment is set up in the development testing environment, utilizing the actual database and integrated services used in product operation. In terms of automation testing, JUnit programs are written to conduct automated regression testing by accessing system interfaces. The test report is generated by a customized testing tool, providing a visual display of the execution results.

5.2 Analysis and Evaluation of Test Results

The system underwent automated testing with 47 detailed functional test cases and 5 performance test cases. The test coverage reached 95%. The test results, as shown in Table 1, indicate that the pass rates for various functional modules are all above 90%, with the core data entry and query modules achieving a pass rate of 96%. The performance test cases primarily focused on system response time, and all of them passed successfully. However, some of the approval workflow test cases did not pass due to the complexity of rule configurations.

Table 1. Pass Rate Statistics for Test Cases

Functional Module	Number of Test Cases	Number Passed	Pass Rate
Entry	20	19	95%
Approval	15	14	93%
Query	12	12	100%
Total	47	45	96%

From the performance test results (see Figure 3), under the simulated condition of 100 teachers simultaneously uploading 3000 sets of grade data, the system, through optimization measures such as index optimization and cache addition, reduced the peak response time from 1.32 seconds to 0.8 seconds, and the average response time from 0.51 seconds to 0.3 seconds. Meanwhile, in a scenario simulating 500 students simultaneously querying grades, the optimized peak response time is 1.2 seconds, with an average time of 0.28 seconds. The performance metrics for key business scenarios all meet the design goal of less than 2 seconds.

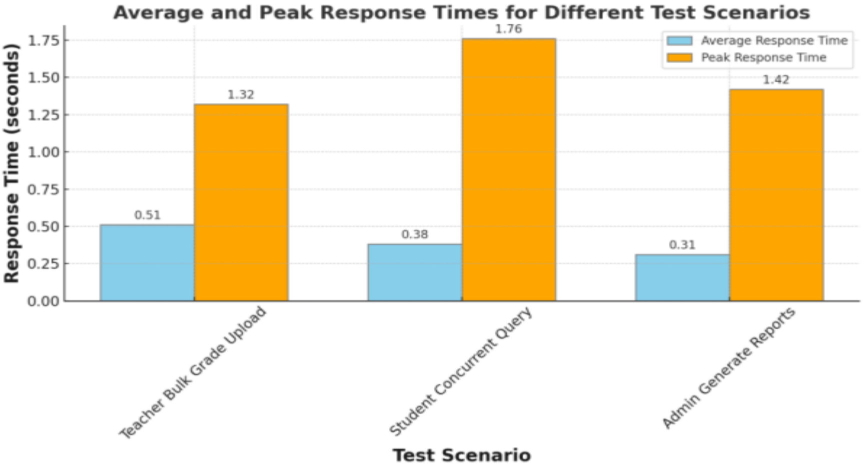
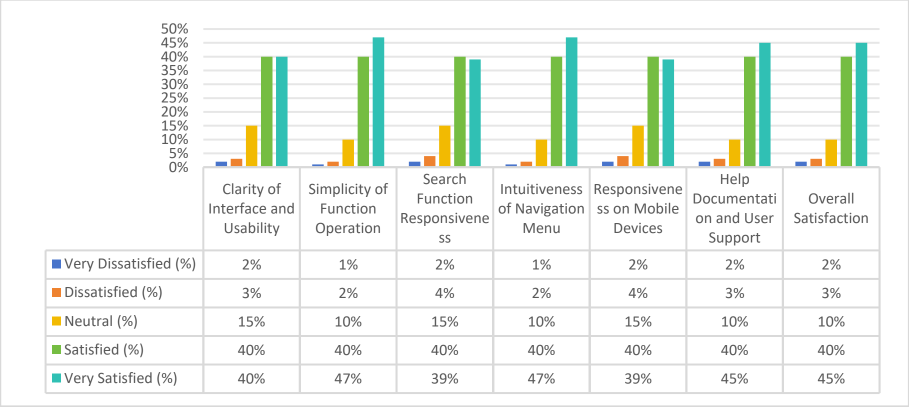


Fig. 3. Average and Peak Response Times for Different Test Scenarios

In addition to functional and performance testing, usability testing and security penetration testing were also conducted. Usability testing results indicate that 80% of users provided positive feedback, stating that the system interface is user-friendly and the functionality is easy to learn and use. Security penetration testing simulated attacks involving 10 different types of abnormal inputs, verifying the system's security protection mechanisms, including input filtering and log auditing, as shown in Figure 4.



	Clarity of Interface and Usability	Simplicity of Function Operation	Search Function Responsiveness	Intuitiveness of Navigation Menu	Responsiveness on Mobile Devices	Help Documentation and User Support	Overall Satisfaction
🟦 Very Dissatisfied (%)	2%	1%	2%	1%	2%	2%	2%
🟧 Dissatisfied (%)	3%	2%	4%	2%	4%	3%	3%
🟨 Neutral (%)	15%	10%	15%	10%	15%	10%	10%
🟩 Satisfied (%)	40%	40%	40%	40%	40%	40%	40%
■ Very Satisfied (%)	40%	47%	39%	47%	39%	45%	45%

Fig. 4. User satisfaction

This testing was a complete success, confirming the system's design objectives and readiness for official operation, providing excellent service to a wide range of users. Subsequent testing will continue to be organized to adapt to evolving requirements.

6 Conclusion

Through the design and implementation process of this system, we have created a comprehensive system for managing sports performance in higher education institutions. The system utilizes a modular and service-oriented architecture, offering multiple functional modules including role-based authorization management, grade entry and approval, data statistics and analysis, and mobile querying. The system supports flexible permission control and efficient batch data processing, significantly enhancing operational efficiency. Complex grade calculation rules have been successfully applied and validated, meeting the needs of educational evaluation. The system exhibits fast response times, strong scalability, and excellent test results. As a crucial infrastructure for managing physical education in higher education, this system has been successfully applied in the evaluation and management of multiple colleges. After a semester of operation, the system has shown stability, and positive feedback has been received from administrators and teacher users. Their suggestions for feature enhancements will guide our next iterations. The project has a high completion rate, good quality, and a favorable cost-benefit ratio. The design and practical development of the system have proven the chosen technological path to be correct. The standardized architecture based on unified identity authentication holds vast application prospects and will further expand the system's scale, promoting the sharing and innovative application of industry data elements.

References

1. Wang T, Park J. Design and Implementation of Intelligent Sports Training System for College Students' Mental Health Education[J]. Frontiers in Psychology, 2021, 12:634978.
2. Yamaguchi S, Inami T, Yamashita D, et al. Physical Characteristics and Performance of Starters and Non-Starters in Elite-Level Female Soccer Players in College: A Case Study of Japanese Athletes[J]. Football Science, 2022, 19:49-58.
3. Cojocaru A M, Cojocaru M, Jianu A, et al. The Impact of Agile Management and Technology in Teaching and Practicing Physical Education and Sports[J].Sustainability, 2022, 14.
4. Cornett V, Gülbahar Urhan. Performance anxiety experiences and coping techniques of Turkish music students and their teachers: [J]. International Journal of Music Education, 2021, 39(4):504-519.
5. Trabelsi O, Gharbi A, Souissi M A, et al. Video modeling examples are effective tools for self-regulated learning in physical education: Students learn through repeated viewing, self-talk, and mental rehearsal: [J]. European Physical Education Review, 2022, 28(2): 341-360.
6. Sisavath C, Yu L. Design and implementation of security system for smart home based on IOT technology[J]. Procedia Computer Science, 2021, 183(2):4-13.
7. Munro A K, Hunter E C, Hossain S Z, et al. A systematic review of the menstrual experiences of university students and the impacts on their education: A global perspective[J]. PLoS ONE, 2021, 16(9):e0257333-.
8. Yadav S, Gulia P, Gill N S, et al. A Real-Time Crowd Monitoring and Management System for Social Distance Classification and Healthcare Using Deep Learning[J].Journal of healthcare engineering, 2022, 2022:2130172.
9. Xie S, Xu J. Design and Implementation of Physical Education Teaching Management System Based on Multi-agent Model[J].International Journal of Computational Intelligence Systems, 2023, 16(1).
10. Fu Q, Lai J, Zhong T, et al. Design and Implementation of Clinical LIS360 Laboratory Management System Based on AI Technology[J].International Journal of Computational Intelligence Systems, 2023, 16(1).