

## Date Input Validation on Score Management of Examination System Based on Struts2

Minghui Li

College of International Exchange  
Bohai University  
Jinzhou, P.R. China  
liminghui0607@sina.cn

**Abstract**—University management modernization and informatization plays an important role in improving the students quality, score management is one of the important links. In order to improve the accuracy of data input of the B/S structure program, prevent illegal input into the system, this paper study on validation framework based on Struts2. Firstly, the research of Struts2 framework validation process and input validation methods; then, carried on the system database design, including the conceptual structure design and logical structure design; Finally, the system used required, stringlength, int, double, date, email. 6 validators are described, and the design of the verification file example. Contents of this paper to facilitate system development, practical application can be used in combination with a regular expression, in order to make the verification method more scientific and effective.

**Keywords**—struts2; examination system; score management; data input validation

### I. INTRODUCTION

With the rapid development of computer technology, network technology and database technology, has been widely used in social economy, politics, military, culture, health and education and other fields, and with the improvement of the level of information, more and more areas are enjoying computer technology, network technology and database technology to bring the advantages of high efficiency, high cost and so on. The quality of learning, work and life in all aspects of the people has been greatly improved and enhanced. Score management is one of the main daily management work of each school, involving schools, teachers, students, and many other aspects, with the reform of the teaching system, especially the development and deepening of the credit system and elective system, daily management work of student score and scores preservation management is increasingly heavy, complex. An urgent need to research and development of powerful, simple to operate, with humane student score management system [1,2].

B/S (Browser/Server) structure integrated browser, information services and Web technology, built on Internet/Intranet technology, has the following advantages: a unified browser, consistent user interface, operation, use and maintenance simple; using the standard TCP/IP, HTTP protocol, can achieve the interconnection between networks, information sharing, system scalability is good; firewall technology can achieve a high degree of information security.

For the convenience of students and teachers to use, score management system suitable for using B/S structure, due to the openness of the Web application, and data collected by the input page is very complex, not only contains the normal user input errors, and may contain malicious user input errors. A robust application system must prevent illegal import of these applications, the prevention of illegal input into the system, so as to ensure the system is not affected. Therefore, input validation is all Web applications must solve the problem, this paper studies score management system input validation based on Struts2 validation framework, and provide support for system development.

### II. STRUTS2 AND ITS VERIFICATION FRAMEWORK

Struts2 with WebWork excellent design ideas as the core, to absorb some of the advantages of the Struts framework, provides a more clean MVC design pattern to achieve a Web application framework [3]. Struts2 framework introduces several new features: separating crosscutting concerns interceptor from the logic, reduce or eliminate the configuration file, a powerful expression language throughout the entire frame, supports change and reusable tag API based on MVC pattern, Struts2 makes full use of the MVC framework from other experiences and lessons learned, making the Struts2 framework more clear and flexible.

#### A. Struts2 Workflow

Struts2 workflow is shown in figure 1.

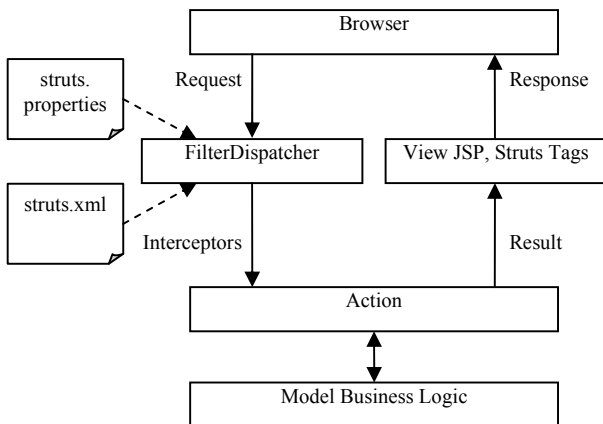


Figure 1. Struts2 workflow

A request in the framework of Struts2 processing steps:

- The client initiates a request directed Servlet container;
- According to the Web.xml configuration, the request first through ActionContextCleanUp filter, which is an optional filter, this filter is useful for the integration of Struts2 and other frameworks, mainly to clean up the current thread ActionContext and Dispatcher;
- Request through the plug filters, such as: SiteMesh, etc filters and other filters ;
- Request through the core filter FilterDispatcher, execute the doFilter method, in this method, ask ActionMapper to decide whether the request to call an Action;
- If ActionMapper decides to need to call a Action, ActionMapper returns a ActionMapping instance (configuration information of storage Action), and the creation of ActionProxy (Action agent) object, the request to the proxy object to continue processing;
- ActionProxy object based on ActionMapping and Configuration Manager to ask the configuration file of framework, find the need to call the Action class;
- when the ActionProxy object is created, it will create an instance of ActionInvocation simultaneously;
- ActionInvocation instance use naming patterns to call, before and after calling the Action process that involves relevant interceptor call;
- once Action execution completed, ActionInvocation instance is responsible for creating and returning Result according struts.xml configuration. Result is usually a JSP or FreeMarker template needs to be represented, it could be another one Action chain;
- What if it want to do before returning Result, can achieve PreResultListener interface, PreResultListener can be implemented in Interceptor, it can also be implemented in the Action;

- According to the Result object information, generate user response information Response, generating a response process can use inheritance tag in the framework of Struts2, in this process will again relate to ActionMapper.

### B. Struts2 Input Validation

Struts2 input validation is divided into client and server validation.

Client validation is through the JavaScript in the data collection page for preliminary filter, JavaScript is a very powerful web scripts, support for regular expressions. The main role of client side validation is to prevent normal browsing error input, can only make a preliminary filtering of the input, for malicious behavior of users, client validation is powerless. Therefore, the client must not replace the server side validation check.

Server validation is on the server side using the program code or configuration files way to verify user input data. Server validation is the last line of entire application defense to prevent illegal data, is realized primarily through the application programming.

Input validation is a presentation layer for data processing, and therefore should be provided by the MVC framework. Struts2 framework provides a very strong input validation system, through the input verifier built in Struts2, without writing any input validation code, you can complete most of the check, and can complete the client and server validation. If the input validation rule application is particularly complex, Struts2 also allows rewriting validate methods to complete the custom calibration; in addition, Struts2 openness allows developers to provide custom validator.

## III. DATABASE DESIGN

Database design is the establishment of a database and application systems technology, is an important part of software design. Specifically, the database design is the application for a given environment, construct the optimal database schema, the establishment of databases and its application system, which can effectively store data, to meet the needs of various user applications [4]. According to the standard design method is divided into six steps of database design, this article only conceptual structure design and logical structure design.

### A. Conceptual Structure Design

Conceptual structure design is the process of user needs will abstract information structure, indicating the raw data in the form of a link between the real world will be reflected in the entities, attributes and entities, and to create a user view of the database [5]. The results of the conceptual structure design are the entity relationship model (E-R, Entity-Relationship).

The examination scores management system relates to the "Student" and "Score" two entities, two entities is the one to many relationship, namely the presence of a Student multi course grades Score, only one course scores corresponding to

a student [6,7]. System entity relationship diagram is shown in figure 2.

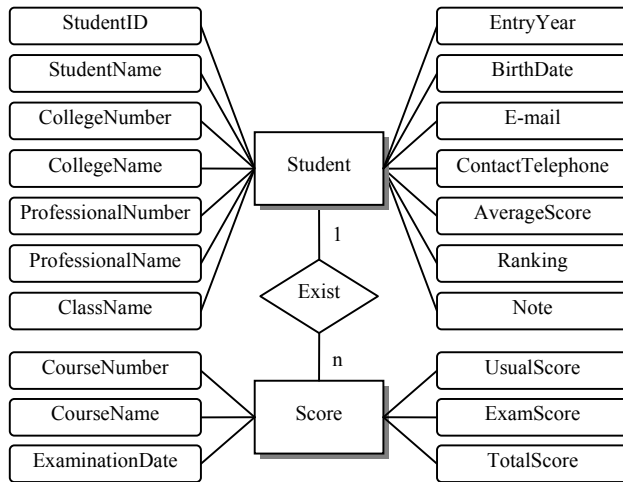


Figure 2. E-R diagram of examination score management system

### B. Logical Structure Design

Logical structure design task is to design the conceptual structure E-R diagram, converted to logical structure is consistent with the a selected database management system products supporting data model. This paper select Microsoft SQL Server 2005 as the database management system.

There are two entities and a "one to many" relationship in figure 2, according to the design rules of logic structure: the transformation of an entity for a relationship model, attribute of the entity is attribute of relationship, the entity code is the relationship code; a "one to many" relationship can be converted into an independent relationship model, you can also merge multipoint corresponding relational schema. If the conversion to an independent entity relationship model, the contact is connected with the code and associated attributes are converted to the relationship attributes between attributes, and the code of relationship is multiterminal entity code. In order to reduce the number of database objects, this paper adopts "one to many" contact with multiterminal merging way. Therefore, E-R diagram shown in figure 2 can be converted into two relational model, which are "Student information table" and "Score information table".

"Student information table" The structure is shown in Table I.

TABLE I. STUDENT INFORMATION TABLE

No	FieldName	DataType	Width
1	StudentID	varchar	10
2	StudentName	varchar	50
3	CollegeNumber	varchar	10
4	CollegeName	varchar	50
5	ProfessionalNumber	varchar	10

6	ProfessionalName	varchar	50
7	ClassName	varchar	50
8	EntryYear	smalling	2
9	BirthDate	datetime	4
10	E-mail	varchar	50
11	ContactTelephone	varchar	50
12	AverageScore	decimal	6,2
13	Ranking	smalling	2
14	Note	varchar	max

"Score information table" The structure is shown in Table II.

TABLE II. SCORE INFORMATION TABLE

No	FieldName	DataType	Width
1	StudentID	varchar	10
2	CourseNumber	varchar	10
3	CourseName	varchar	50
4	ExaminationDate	datetime	4
5	UsualScore	decimal	6,2
6	ExamScore	decimal	6,2
7	TotalScore	decimal	6,2

### IV. VALIDATIONDESING

The process of verification design is to design verification documents according to the validator.

#### A. Validator Explanation

In the Struts2 each Action separately for specify a checksum file, checksum file has the similarity, limited by the length of the article, the paper design in Table I, as an example to design verification document. Table I 14 fields are summarized, the need to design 6 categories validator [8,9].

- Verification is not empty. For completeness of information, usually in Table I. All the fields are not empty. However, when the entry personnel input data, some data may be collected or not certain data is uncertain, temporarily can be empty. Therefore, according to the actual situation to determine program design. Check the type name is "required". The validator can accept one parameter: fieldName, specified the Action class name of verification.
- Verification string length. All of the data type is the field of varchar and char type, if exceed the length set in a database table, there will be errors when stored, and therefore need to be string length check. The validation requirements are checking the length of the field must be within the specified range. Validation type name is "stringlength". The verifier can accept four parameters: fieldName, designated check Action class name; maxLength, the maximum

length of the string, the parameter is optional, if not specified, the maximum length is unlimited; minLength, specified the minimum length of string, the parameter is optional, if you do not specify, a minimum length is unlimited; trim, whether in front of a blank check truncation is property values before and after calibration, this parameter is optional, the default is true.

- Verification integer. The value of verification field required to be within must be in the specified range. Verification type name is "int". The verifier can accept three parameters: fieldName, specified verification Action class name; min, the minimum value of the specified property, this parameter is optional, if not specified, no checking minimum; max, the maximum value of the specified attribute, the parameter is optional, if not specified, the maximum value is not checked.
- Verification double precision floating point number. Requirements value within the specified range, and verify that the Action property is double precision floating point number. Verification type name is "double". The validator can accept three parameters, and "int" validator is same.
- Verification date. Require field value must be within the specified date range. Verification type name is "date". The verifier can accept three parameters: fieldName, specified verification Action class name; min, the minimum date of specified attributes, this parameter is optional, if not specified, no checking minimum date; max, the maximum date of the specified attribute, this parameter is optional, if not specified, no check the maximum date.
- Verification e-mail address. If the check field characters is non-null, it must be a legitimate email address. Verification type name is "email". The validator can accept one parameter: fieldName, designated check Action class name. The verifier verifies based on regular expressions, with the development of computer technology, system settings the regular expression may not fully cover the actual e-mail address, in order to ensure the correctness of the program permanent, it is best to use regular expressions complete mailing address verification.

## B. Validation File Desing

Struts2 Verification file named as <Action-name> - validation.xml, stored in the same package with the Action, verification only need to configure a verification file can be realized. Six categories described above, each validator gives an example, verification file design is follows:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE validators PUBLIC "-//OpenSymphony
  Group/XWork Validator 1.0//EN"
  http://www.opensymphony.com/xwork/xwork-validator-
  1.0.2.dtd" >
<validators >
```

```
<!-- Validation "Non empty" example-->
<field name="StudentID">
  <field-validator type="required">
    <message>StudentID cannot be empty</message>
  </field-validator>
</field>
<!-- Validation "String length" example -->
<field name="StudentName">
  <field-validator type="stringlength">
    <param name="minlength">2</param>
    <param name="maxlength">50</param>
    <message>StudentName must be between 2 and 50
      characters</message>
  </field-validator>
</field>
<!-- Validation "Integer" example -->
<field name="Ranking">
  <field-validator type="int">
    <param name="min">0</param>
    <param name="max">9999</param>
    <message>Ranking must be between 0-9999</message>
  </field-validator>
</field>
<!-- Validation "Double precision (floating-point )" example
  -->
<field name="AverageScore">
  <field-validator type="double">
    <param name="min">0</param>
    <param name="max">999.99</param>
    <message>AverageScore must be between 0-
      999.99</message>
  </field-validator>
</field>
<!-- Validation "Date" example -->
<field name="BirthDate">
  <field-validator type="date">
    <param name="min">1949-01-01</param>
    <param name="max">2049-12-31</param>
    <message> BirthDate must be between ${min}-
      ${max}</message>
  </field-validator>
</field>
<!-- Validation "Email address" example -->
<field name="E-mail">
  <field-validator type="email">
    <message>E-mail address format is
      incorrect</message>
  </field-validator>
</field>
</validators>
```

## V. CONCLUSION

Teaching is the core work of the school management, modern management tools objectively requires. Through student scores management system integrated with the school management, teaching can improve the teaching management level, is an effective means of promoting the modernization of school management. Through the system construction, to achieve student management digitization, to adapt to the development trends of information, to meet the full-time undergraduate teaching characteristics, to facilitate

the students and teachers; for the realization of modernization, standardization of teaching management, to achieve the "paperless" office lay a foundation, to improve the quality and efficiency of education management. Struts2 is a popular MVC framework, data input validation is one of the core technologies of Struts2 framework. In order to improve the efficiency and quality of examination scores management system developed and expanded research. In this paper, according to the student information table using the required, stringlength, int, double, date, email and other validators, you can choose the most appropriate calibration according to the actual needs of the application, and combined with the use of regular expressions validation methods, to make validation methods is more scientific, more secure data input.

#### REFERENCES

- [1] J. Li, Y. J. Chen, X. B. He, "The Design and Implementation of Performance Management System Based on Struts2 Architecture," *Journal of China West Normal University(Natural Sciences)*, vol. 35, no. 3, pp. 281-285, 2014.
- [2] C. Q. Li, "The design of student achievement management system based on SSH2," *Computer Knowledge and Technology*, vol. 10, no. 21, pp. 4955-4958, 2014.
- [3] Z. G. Zhang, "Development and application of Struts2 framework technology," *Computer CD Software and Applications*, vol. 16, no. 23, pp. 119-120, 2013.
- [4] Luob blog, "Database Design," [http://www.360doc.com/content/11/0803/10/1039473\\_137600904.shtml](http://www.360doc.com/content/11/0803/10/1039473_137600904.shtml), 2015-1-15.
- [5] P. Wei, "Database design I see: to database development of a university teaching evaluation system as an example," *Journal of Changsha Railway University(Social Science)*, vol. 15, no. 2, pp. 289-290, 2014.
- [6] Z. W. Yang, "Designation and implementation of College Student Achievement Management System," *Master's Degree of University of Electronic Science and Technology of China*, 2013.
- [7] Z. Wang, "The Design and implementation of Student Achievement Management System," *Master's Degree of University of Electronic Science and Technology of China*, 2013.
- [8] X. M. Li, "Based on the Input Validation Functions Struts2 Framework Implementation and Application," *Computer Knowledge and Technology*, vol. 9, no. 35, pp. 8201-8202, 2013.
- [9] Luob blog, "Struts2 use the Validation framework to validate the data (thirty-seven)," <http://takeme.iteye.com/blog/1667716>, 2014-12-15.