Research on Servlet Technology in Personal Financial Management System

[I]Zhou Jie, [II]Zhang Yongbo

[I]Shandong University of TechnologySchool of Comp. Sci. and Tech., Shandong Zibo255049;
[II]Shandong Zibo Information Engineering School, Shandong Zibo255000

Abstract

Server-side programming methods JSP and Servlet have become the preferred way to develop computer applications today. This paper describes the filter technology on the J2EE platform, analyzes the basic working principles of JSP, Servlet, and the interaction process between components. In the personal financial management system under the MVC architecture, Servlet + JSP is used to implement the module design. Strengthen the information security protection mechanism of the system.

Key words

MVCarchitecture;controller;Personal finance;Servlet;JSP

I. Introduction

With the advent of the Internet and the era of big data, personal finance has become quite common in many places. If you use computers to manage personal finance, it has the advantages that traditional manual statistics do not have. For example: simple statistics, simple search, strong reliability, high confidentiality. The huge market demand has prompted us to study the personal finance system.This paper presents a design scheme of a personal financial management system based on filter development technology. The personal financial management system is a modern management software that allows users to record, add, modify, and delete their own income, expenses, and various information through the use of computers.The personal financial management system uses the MVC architecture, selects JSP, Servlet, JavaScript, etc. as the application development language, and selects SQL Server as the database for the background.

II. MVC Architecture Introduction

MVC(Model-View-Control) is a software design model invented in the 1980s. The MVC architecture divides the interactive system components into three parts: model, view, and controller.Model is an object model abstracted from the real world. The model encapsulates the number and operation of data, and performs data processing calculations.View is the interface between the application and the user and is responsible for rendering the application to the user.Controller is responsible for the interaction between the view and the model, controlling the response, response, and process to the user input. It is mainly responsible for:(1) distributing the user's request to the corresponding model. (2) The changes to the model are reflected in the view in a timely manner.

Using the MVC design idea, the three-tier architecture is implemented to separate the display layer from the logical layer, so that a model can have multiple display views. When the model data changes, the controller notifies all views that the display needs to be updated.At the same time, the separation of data layer and presentation layer is realized, and no changes are required when replacing or updating one module. The use of MVC architecture has greatly improved the efficiency of program development.

III. JSP Technical Analysis

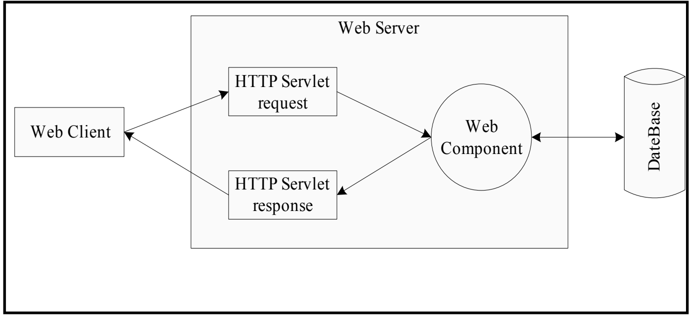
JSP + Servlet technology is the basis of the J2EE(TM) platform, which provides connections between Web clients and server-side programs.Servlet is nested in Java HT – ML. It is essentially a Java program that has all the advantages of the Java language including portability, reuse, and crash protection. JSP is a server-side extension of Servlet technology. The JSP engine is included in the Servlet engine. Before the program is executed, JSP leads the prime to convert / compile the JSP page to Servlet.The client sends a message request. The JSP engine calls the Java compiler to convert the JSP file into a Servlet source file with the suffixname of the file.Java, then compile it into a class file with the suffix name of the file.Class. Finally, the Java virtual machine explains and executes the class file.

IV. Servlet interaction process

Servlet is a Java application. Servlet interacts with the Web browser through the HTTP protocol. After the interaction is completed, Servlet returns the generated response to the Web server. The main function of Servlet is to browse and modify data interactively to generate dynamic Web content. Its workflow flow chart is shown in the figure.

Servlet Workflows :

1. The client sends the request information to the server.
2. The server sends the request information to Servlet.
3. Servlet generates a response to pass it to Servlet. The dynamic generation of response content generally depends on the client's request.



Servlet workflow flowchart

4. The server returns the response information to the client.
5. Implementation of JSP + Servlet under MVC Architecture

When MVC design ideas complete Web application development, JavaBean and Servlet form a business logic layer(Service layer), which is the JSP + JavaBean + Servlet implementation mode. In this implementation mode, the JSP page is responsible for receiving requests from users and the display of data. The

JavaBean component implements business logic. The Servlet component is used to perform business logic and is responsible for the process control of the program. The three roles are clear and the division of labor is clear. It is a concrete implementation of the MVC, the Model-View Control design model, and it is an implementation model that must be mastered in the process of learning JSP development technology.

Taking the Web user login program in the personal financial management system as an example to illustrate the main role of the MVC design model in actual development.The user login function is one of the basic functions that the personal financial management system needs to realize. Design a simple user login personal financial management system.In this procedure,the user input login information is submitted to the Servlet. After the Servlet receives it, the legitimacy of the input content is first checked. If the verification fails, the error information is passed to the login page display. If the data is valid, call the DAO layer to complete the validation of the database and jump to the page where the login is successful or failed according to the result of the verification.

The MVC login list is shown in table 1.

Table 1 : MVC Logon Program List

No.	Name	File Type	Describe
1	User	JavaBean	VO operation class for user login
2	Database Connection	JavaBean	Complete database opening and closing operations
3	IUserDAO	JavaBean	Define the DAO interface for the login operation
4	UserDAOImpl	JavaBean	Real implementation class, complete specific login verification
5	UserDAOProxy	JavaBean	Responsible for opening and closing databases, calling real topic classes
6	DAOFactory	JavaBean	Gets the instantiated object for the DAO interface
7	LoginServlet	Servlet	Receive request parameters, validate parameters, return login information
8	login.jsp	JSP	Provides a user input form showing the user's login success or failure

Using the Mysql database, the database name mvcsql, table name is: user, which contains three attributes, namely:(user, name, password).

Here's the code for part of the program.

Some of the model classes have the following components.

1. DatabaseConnection class.

```
public class DatabaseConnection {
 // This class is mainly used to complete database opening and closing operations.
……
publicDatabaseConnection() throws Exception {
try {
Class.forName(DBDRIVER); // Load Driver
this.conn = DriverManager.getConnection(DBURL, DBUSER, DBPASS);
} catch (Exception e) {
……
}
}
 PublicConnectiongetConnection() { // Get a connection to the database
……}
 }
```

2. IUserDAOclass.

```
public interface IUserDAO { // defines the interface for the DAO operation;
// User login validation
publicbooleanfindLogin(User user) throws Exception; }
```

3. DAOFactory class.

```
public class DAOFactory { // Gets the instantiation of the DAO interfaceobject
public static IUserDAOgetIUserDAOInstance()throws Exception {
return new UserDAOProxy(); // Get an instance of the proxy class
}
}
```

4. login.jsp.View Display

```
<form action=”LoginServlet” method=”post”onSubmit=”return validate(this)”>
……
</form>
```

5. LoginServlet.java: Control class, responsible for forwarding requests, processing requests.

```
public class LoginServlet extends HttpServlet{
publicvoid doGet(HttpServletRequestreq, HttpServletResponseresp) throws ServletException, IOException{
……
if(info.size()==0){
User user = new User();
user.setUserid(userid);
user.setPassword(userpass);
try{
if(DAOFactory.getIUserDAOInstance().findLogin(user))
{
info.add(“User login successful, welcome “ + user.getName() + “!”);
} else {
info.add(“User login failed, wrong username and password!”);
}
……
}
```

V. Conclusion

This paper analyzes the development of Web application with

MVC model combined with Web example. The solution of the program, the MVC model is a very effective model to solve the development of Web applications. It provides a unified and standard development model for Web application development, separating the data objects, business logic, and user interface in the web system. The system has a good expansibility, shortened the development cycle, and facilitated the later maintenance. It will still be widely used today.

References

[1] SunWeiqin. Tomcat and Java Web Development Technology [M] . Beijing: Electronic Industry Press, 2010.

[2] Wang Wenxin. Web application framework design based on model-view-controller[J]. Information and Electronic Engineering, 2009, 7(4): 358-360.