

Efficient Software Architecture through Prototyping Approach

Digambar Y. Yedage, Anand Bone

Abstract— Most of the software industries are focusing on the saving cost in the area of the development and maintenance of the project. The focus is on analysis of JSP, Struts, Spring MVC and Hibernate frameworks for development of the application and the prototype. As per the customers requirement developers has to perform the development task within specified time. Once the software project requirements are clear from the customer, the requirement analysis phase is carried out before the start of the design. In the design phase, the developer is not aware of what are the technologies getting involved. Before starting of the actual development phase, customer wants to see the prototype of an application. This needs to be show cased as the prototype of the application. The prototype development is the predevelopment phase which is an important phase in the software development life cycle. The customer will approve the prototype and then actual development will start. This is the most efficient way of developing the software architecture.

Index Terms— Prototype, Efficient software architecture, Software Architecture, JSP Custom tag and Hibernate.

I. INTRODUCTION

Most of the software industries are in the area of software development of the web applications. All the Information Technology industries are following the software development life cycle for developing an enterprise application. This software development life cycle involves requirement gathering, requirement analysis, design, coding, testing and maintenance. As an important step of the SDLC it is necessary to understand the requirement from the client. Once the requirements are very clear from the client, it is necessary to prepare the design of the application. The design phase of the application is independent of the language to be used during the development. Before starting of the development it is necessary to develop the prototype. During the development of the prototype the developer is not having any idea on what kind of database they will be using, whether it is an ORACLE or CYBASE or SQL server. The developer will have only an idea of the table and column which are going to be used in the development. Before starting of the actual development work the developer need to showcase the prototype which includes the look and feel of the application. The software developer will have very little time for developing these web pages as prototype user interface.

Initially Java Server Pages were used to develop web pages provided by sun Microsystems. To use the JSP we need to use the scriptlets to embed the java code inside the JSP page.

Manuscript received on October, 2013.

Digambar Y. Yedage, Department of Computer Engineering, SKN Sinhgad Institute of Technology & Science, Kusgaon, Lonavala, University of Pune, Maharashtra, India.

Anand Bone, Department of Computer Engineering, SKN Sinhgad Institute of Technology & Science Kusgaon, Lonavala, University of Pune, and Maharashtra, India.

By embedding the java code inside the JSP will make the JSP page as dynamic one. Most of the JSP Page developers or UI developers need to understand the scriptlet code along with the JSP tags. The sample code with scriptlet is given below. Writing the HTML code with java code embedded in it is very difficult to maintain. As per the software development life cycle more time is being spent on the maintenance. To debug the code written with such type of JSP and java code is difficult.

The below example is of writing a web page in JSP with java code written in the scriptlet. Only to print today's date and time, developer needs to understand the java code with HTML knowledge as well.

Java Code for date written in below:

```
java.util.Date today = new java.util.Date();
<html>
<head>
<title>JSP PAGE</title>
</head>
<body>
<p>Welcome to the world of Web programming</p>
<!-- Scriptlets code begins here -->
<%
java.util.Date today = new java.util.Date();
%>
<!-- Scriptlets code ends here -->
<p>Today's date and time now :
<!-- Scriptlets code begins here -->
<%= today%>
<!-- Scriptlets code ends here -->
</p>
</body>
</html>
```

The above JSP and scriptlet code is difficult to understand for the UI developer who has no knowledge on Java. Also writing the java code inside the JSP code doesn't look good for maintenance as well.

To develop the prototype with the help of Hibernate and Java server pages, one way is to write the hibernate code in the scriptlet and then use those JSP's to build the user interfaces. The Hibernate is the ORM framework which generates the query by itself with independent of database used [4].

Hibernate is developed by JBOSS community as an Object/Relational mapping framework. Hibernate framework has different tags as to perform database related operation methods as well [1]. Here in the Hibernate we have a load method to fetch the records from any database and populate it as model which mapped in the ORM mapping. Hibernate is used in small scale to large scale web application development.

The framework proposed in this paper is to access the data persistence (Hibernate) layer in the view layer i.e. JSP layer

for developing small scale and medium scale application to develop any web page, firstly the Java Server Page is written and then to use the Hibernate inside the JSP, used the different custom tags. As in case of MVC design architecture the data has to flow from view, controller and then the model use to get updated by calling the data persistence layer. The approach of MVC takes longer time, for developing the small and medium scale web applications. The development of web application with proposed framework does not require interacting with the persistence layer. The tag library supports different types of Java Server Page tags for executing any Data Access Object layer interaction.

II. BACKGROUND AND RELATED WORK

There are many frameworks available in the market for developing the web application. The web application can be developed with struts, spring and Hibernate combination of framework. Only with the help of struts and hibernate also there are many applications but so as to develop the applications in MVC design pattern takes more development time. If the application is very big and takes time for maintenance then using the using proper architecture and design helps. Whenever there is a need to develop the small or medium scale web application then going through all there architecture and all takes more time. To develop the application very faster and easier way the different frameworks are built. The Struts has more advantages over the traditional JSP with scriptlets [15].

The MVC can be defined as a set of structures needed to reason about the software systems, which comprises of software elements between them. The MVC pattern is fundamentally used to lose coupling between the data with visual representation and manipulation of the data. In MVC the data is going to be the model, the visual representation of the data is the view and the data manipulation is the controller. The controller gives the commands to the model to update the models state. It can also give commands to its associated view to change or update the views presentation of the model. A model notifies its associated views and controllers when there has been a change in the state. This notification allows the views to produce updated output, and the controller to change the available set of commands. A passive implementation of MVC omits these notifications, because the application does not require them or the software platform does not support them. A view can communicate by sending request information from the model that it needs for generating an output representation to the user.

JSP and servlet can be used as to develop the MVC architecture. The view layer will have the JSP code along with the java code embedded in the scriptlet. This layer is responsible for display of the model in the UI. The servlet can be used as a controller. Servlet is responsible for taking the input from the user and it analyses it and then interacts with the model. Also it tells the model to get the updates from database and update its state and then redirects to the view layer. Model layer here is plain old java code with holding the actual data from the database. This is important to get the updates in the view layer.

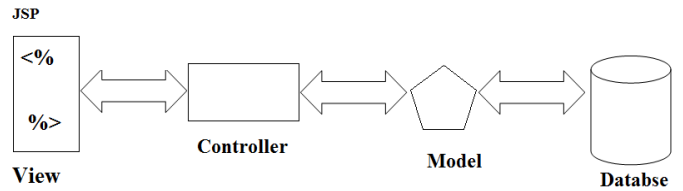


Fig. 1 MVC architecture in JSP

The Struts has more advantages compared with development using JSP. Struts use the logic tag to avoid conditional statements which is required to be written in the JSP. There are different Struts tags being used to avoid the scriptlet code to be written in the JSP. If the size of the application is huge which has more complexity then using the Struts MVC model will help in development and maintenance of the application. Whenever there is a need to develop a small scale application that time it is not necessary to develop it in Struts which would require more time [2].

Most of the application follows the MVC design pattern. The MVC design architecture takes lot of time in the development. As this architecture takes to write model which holds the logic, view for display. To interact efficiently between model and view there is a need to write the controller configuration in xml configuration files.

Model is responsible for holdings the real business logic. Also model takes the data from the data base and sends it to the view layer.

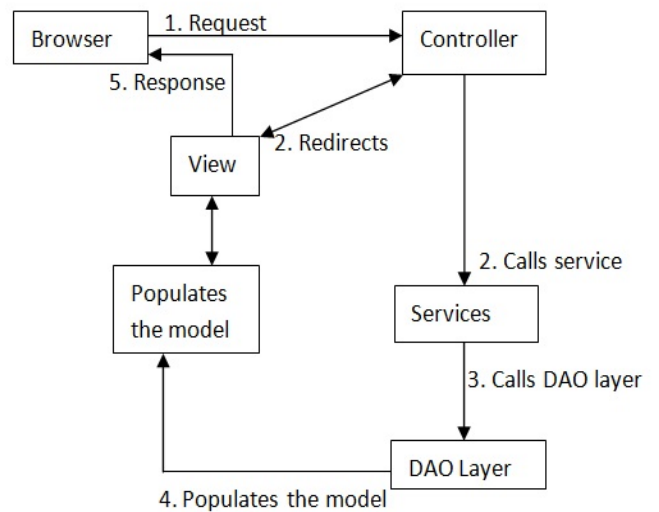


Fig. 2 MVC Design Architecture in Struts

Spring MVC also works on the model – view – controller architecture. The spring’s web Model View Controller has the DispatcherServlet to dispatch the request to the handlers. This framework is request driven. The main components of the Spring MVC are front controller, Controller and view template. Here the DispatcherController is the front controller. Controller interacts with the service layer for processing a business and returns the model. It is responsible for handling and the navigation logic to the business objects. View layer is a just the JSP page written as a user interface. It renders the output model on the UI. Model is a domain object that is processed by a service layer or persistent layer. View Resolver is responsible for how the output is to be displayed depending upon the result received from the Model and View.

III. PROBLEM DEFINITION

The terms software architecture intuitively denotes the high level structure of the software systems. The software prototyping term refers to the activity of creating prototypes of the software applications that is incomplete version of the software program being developed as a final product. It is an activity that mostly occurs in the software development. A prototype simulates a few aspects of the major application, and may be completely different from the final product. In case of small scale application it can be developed in the same fashion.

With the help of this kind of prototype development, the software developer can get the valuable feedback from the customer in the early stage of the project development phase. The customer and the contractor can compare the software prototype developed matches the software specification, according to which the software program is built. It also allows the software developer some insights into the accuracy of the initial project estimates. With the help of this initial step the software developer can give proper estimates so that the deadlines and milestones proposed can be successfully met. The degree of correctness and the techniques used in the prototyping have been in the development.

The purpose of the prototype is to allow customers of the software to evaluate developer's proposals for the design of the eventual product by actually trying them out, rather than having to interpret and evaluate the design based on the descriptions. Prototype can also be used by the end users to describe and prove the requirements have not considered, and that can be a key factor in the commercial relationship between developers and their clients. The prototype can also avoid the great expenses and difficulty of changing the finished software developer's final product.

This paper suggests an approach to design the web application with small scale as well as medium scale in a quicker and faster way. The interaction of view layer with other layer is minimized. Here the access of data persistence layer is available in the view layer. With the help of JSP custom tag libraries, we are planning to develop different types of tags to interact with the Dao layer. With the help of simple tag it is possible to interact with DAO Hibernate layer and fetch the records directly in the Java Server Pages.

IV. PROPOSED METHODOLOGY

Whenever the frameworks are used for the application development the development time is reduced. For example the use of hibernate framework for DAO layer takes very little time in developing because there is no need to write the structured queries languages queries which were used in traditional databases. Generation of the queries is taken care by the framework. Following are some benefits of the proposed approach.

A. Requirement for developing the prototype of the project

The prototype application needs to be developed in a faster mode. To decide the technology stack which will take more time? So it is architecture development of application.

B. Developing application where the application architecture is not important

To develop the small scale application there is no need to design the architecture of the application. As the design of

architecture takes more time and cost hence generally the small scale application does not repair the architecture. [1]

C. Faster development of the small scale application

Title Easiest way of developing the small scale application is to combine the layered architecture into common JSP & develop the application. [2]

D. Learning curve for the view layer developer.

As here we are forced on the JSP & Hibernate combination, there is a huge learning curve for the JSP developer. [1]

The prototype is model shown in the fig. 3 is the model of a proposed product or a service or a system. The prototyping is the process of building a model that demonstrates the features of a proposed product or the systems. Also the prototype can be a proof of concept to prove the technical feasibility of a proposed system. The prototyping phase involves basic four steps.

1. Identifying the basic requirements of an application. This step will help in understanding the requirement from the customer.
2. As per the requirement an initial prototype will be developed by the software developer.
3. In this step the user review or customer review with their review comments will be provided. This review comments will be with respect to the software requirement. If there is any missing functionality then customer will provide the review comments for further update of the prototype.
4. Implement the review comments given in the step 3. By implementing customer or user review comments this updated prototype will again sent to the customer for review process.
5. If the prototype is ok by the customer then it is being approved by the customer.

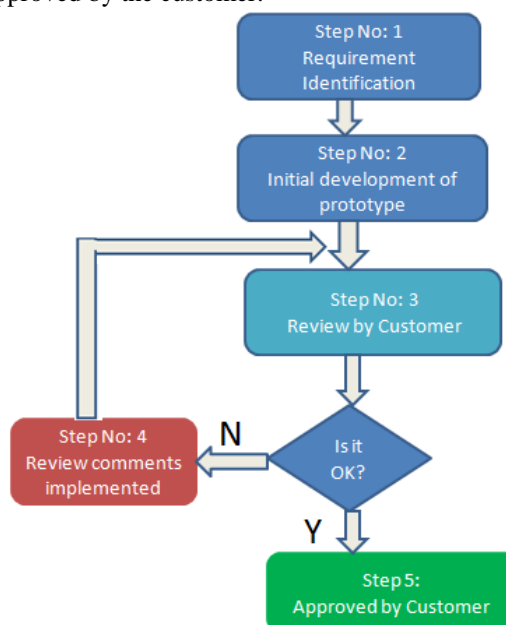


Fig. 3 Prototype model

In the proposed approach, main intention is to save the cost of the project development by initial implementation by prototype development. This approach is going to be suitable for developing the new custom tags which would help any developer to access DAO directly in the view layer also. In case of small scale application, it can be developed using the

proposed approach. To give access to the data access object layer directly through JSP custom tags along with the hibernate framework [1]. The Hibernate framework helps in managing the DAO layer very efficiently.

V. CONCLUSION

As many of the software industries are looking out for saving the development cost of the project. The proposed approach is going to help to minimize the cost of the project. Without having much knowledge on the other technologies the UI developer can develop the application very efficiently. The JSP custom tags will definitely help in developing the prototype which is going to be considered as very important phase in the software review process of the customer.

ACKNOWLEDGMENT

The Authors are thankful to Prof. Anand Bone and Prof. V.D.Thombare for the continuous guidance and support.

REFERENCES

- [1] Efficient Access to Hibernate Through JSP Powered by a New Tag Library, Pubudu Gunawardena, 99X Research, 99X Technology, 2012
- [2] Transforming Embedded Java Code into Custom Tags, Shannon Xu and Thomas Dean, 2012
- [3] H. Bergsten. Java Server Pages. O'Reilly Media, 2003, pp. 1-28
- [4] JBoss Community. Relational Persistence for Java and .NET. [Online]. <http://www.hibernate.org/>
- [5] Pugh and J.D. Gradecki. Professional Hibernate. Wiley, 2004, pp. 15-22
- [6] C. Bauer and G. King. Java Persistence with Hibernate. Manning, 2009, pp. 4-10
- [7] B. Basham et al. "Script-free pages" in Head First Servlets and JSP. O'Reilly Media, 2004, pp. 341-348
- [8] A. Hasegawa et al., "BioTags: A JSP Tag Library for Bioinformatics Website Construction," Genome Informatics, Japan, Rep. 553-554, 2002.
- [9] Serge Knystautas. Hibtags. [Online] <http://hibtags.lokitech.com/index>
- [10] B. Basham et al. "Custom tags are powerful" in Head First Servlets and JSP. O'Reilly Media, 2004, pp. 435
- [11] B. Basham et al. "When even JSTL is not enough" in Head First Servlets and JSP. O'Reilly Media, 2004, pp. 489-547
- [12] X. Guo, J.R. Cordy and T.Dean, "Unique Renaming of Java Using Source Transformation", IEEE 3rd International Workshop on Source Code Analysis and Manipulation, Amsterdam, p151-160, September 2003.
- [13] F. Ricca, P. Tonella, and Ira D. Baxter, "Web Application Transformations based on Rewrite Rules", Information and Software Technology., vol. 44, n. 13, pp. 811-825, October 2002
- [14] N. Synytskyy, J.R. Cordy and T.R.Dean, "Resolution of Static Clones in Dynamic Web Pages", Proc. WSE 2003, IEEE 5th International Workshop on Web Site Evolution, Amsterdam, p49-58, September 2003.
- [15] S. Xu and T. Dean, "Modernizing Java Server- Pages", being submitted to WSE 2005



Prof. Anand Bone is born in Karnataka, Inida. He has received Master of Engineering in Information Technology from AMC Engineering College Bangalore, Karnataka. Currently he is working as Assistant Professor in Computer Engineering at SKN Sinhgad Institute of Technology and Science, Kusgaon, Lonavala, University of Pune, Maharashtra, India.



Digambar Y. Yedage is born in Solapur, India, in 1985. He has received the Bachelor of Engineering degree in Computer Science and Engineering from Shivaji University, Kolhapur in 2007. He has achieved 7th rank in Shivaji University Kolhapur in B.E. (CSE). Currently he is pursuing his Master of Engineering in Computer Engineering at SKN Sinhgad Institute of Technology and Science, Kusgaon, Lonavala, Pune from University of Pune, Maharashtra, India.