
Software Project Architectural Approach using Java Struts

Ankur Saxena* and Ankur Chaurasia
Amity University, Noida, 201303, India.
*asaxena1@amity.edu

Abstract

This type of research represents an architectural procedural approach to design a Web application using Java Struts framework. This framework is used to develop MVC, POJO and ONGL technique to separate business logic and presentation logic. Struts is part of J2ee simplify the foundation of the enterprise level application program, also cause the designers and the programmers to distribute the function in each discreteness of the server end when using J2ee to establish the application programs. Our research study show that applying multiple frameworks to design the Java applications using MVC,POJO,ONGL concepts makes applications easier compare to a single Framework. In recent trends, more research papers have been proposed, which employs a new and fast process to implement Web architecture and to avoid Framework.

Keywords - Java, J2ee, Struts1, Struts2, MVC, POJO, ONGL, Web, Framework, ValueStack

Introduction

J2EE (Java 2 Platform, Enterprise Edition) is Sun's preferred Java platform for multi-tier enterprise applications. It simplifies enterprise applications by basing them on standardized, modular components, by providing a complete set of services to those components, and by handling many details of application behaviour automatically, without complex programming.

J2ee uses a multi-tier distributed application model. There are three tiers in the typical J2EE application model: Web presentation tier, business logic tier and data tier (Wang *et al.*, 2009).

Web is the very complex issues these days. Since the desire of the companies and organizations are increasing so the complexity and the performance of the Web programming matters. Complexity with the different types of communication devices is increasing (Garg *et al.*, 2011). The business is demanding applications using the Web and many communication devices so that many of the companies use frameworks for making the development of their applications easier. The business today demands Web applications to advertise its company so it is very important to take care of the architecture used in development of the application. Framework can be considered as a set of functions helping the developers in creating the applications (Gupta *et al.*, 2010). So with the increase load of the data on the internet we have to take care of the architecture issue (Garg *et al.*, 2011).Let us discuss in brief the research done of Struts2 framework.

MVC: Model-View-Controller ("MVC") is architectural design pattern for interactive applications. MVC organizes an interactive application into three separate modules (Wojciechowski *et al.*, 2004).it is a design pattern originally pioneered in the olden days of Smalltalk. Successful use of the pattern isolates business logic from user interface considerations, resulting in an application where it is easier to modify either the visual appearance of the application or the underlying business logic without affecting the other. The MVC divides the application into three associated parts: Model, View and Controller, which makes the system development simple (Dey, 2013). Controller: Handles navigation logic and interacts with the Service tier for business logic2. Model: The contract between the Controller and the View Contains the data needed to render the View populated by the Controller 3. View: Renders the response to the request Pulls data (Gupta *et al.*, 2010).

In this architecture (Figure 1.2) browser send a request to jsp and jsp check setter and getter method inside JavaBean. Bean map the data from database and finally jsp send a response to browser (Saxean, 2012).

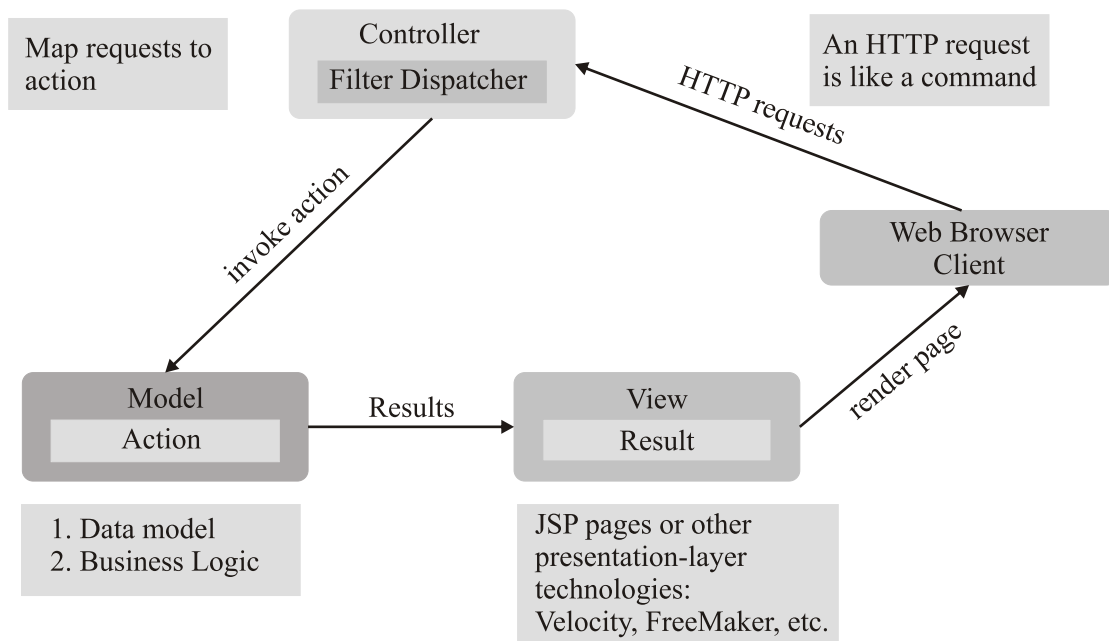


Figure 1: MVC Architecture.

Struts1

Struts are a Framework that implements a powerful and flexible controller based on the Service to Worker pattern. Struts' main advantages are: Integration flexibility: Struts' architecture provides flexibility for choosing the view and the model to be used. The view is based on the plug-ins concept (Brown, 2011).

Struts1 is an open source Framework that extends the Java Servlet API and employs a Model, View, Controller (MVC) architecture. It enables you to create maintainable, extensible, and flexible Web applications based on standard technologies, such as JSP pages, JavaBeans, resource bundles and XML.

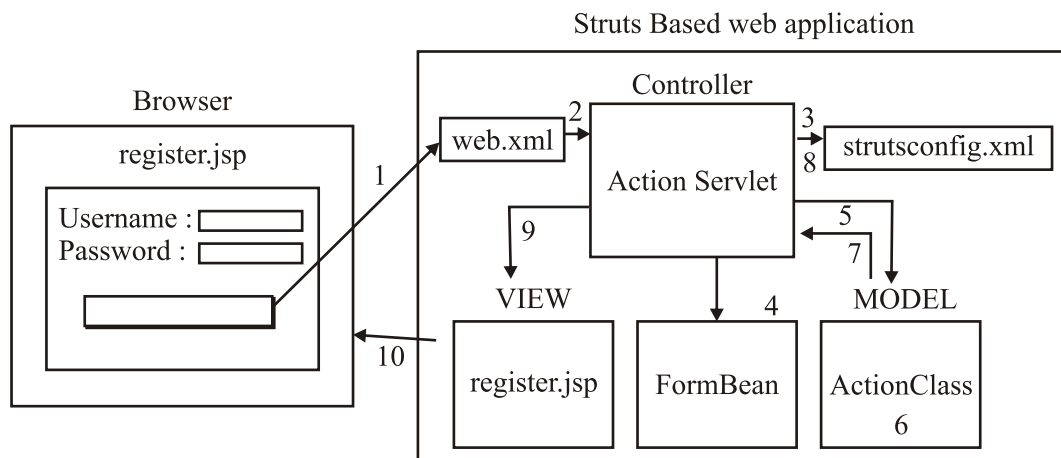


Figure 2: Architecture of Struts1 framework.

Struts2

Struts2 provides supports to POJO based actions, Validation Support, AJAX Support, Integration support to Hibernate and spring frameworks, support to various result types such as Velocity, JSP etc.

In Struts2, action class is POJO (Plain Old Java Object). You are not forced to implement any interface or extend any class. Only execute method should be specified that represents the business logic. The simple action class may look like:

```
Ankur.Java
package ank;

    public class Ankur
    {
        public String execute()
        {
            return "success";
        }
    }
```

A Struts2 provides many features that were not in Struts1. The important features of Struts2 framework are given below:

- An Action class implements an Action interface. Struts2 provides a base ActionSupport class that implements commonly used interfaces. Although an Action interface is not necessary, any POJO object along with an execute signature can be used as a Struts2 Action object.
- Struts2 Actions are not coupled to a container. Most often the servlet contexts are represented as simple Maps, allowing Actions to be tested in isolation. Struts2 Actions can still access the original request and response, if required. However, other architectural elements reduce or eliminate the need to access the HttpServletRequest or HttpServletResponse directly.
- Struts2 uses Action properties as input properties, eliminating the need for a second input object. Input properties may be rich object types which may have their own properties. The Action properties can be accessed from the Web page via the taglibs. Struts2 also supports the ActionForm pattern, as well as POJO form objects and POJO Actions. Rich object types, including business or domain objects, can be used as input/output objects. The ModelDriven feature simplifies taglib references to POJO input objects
- Struts2 supports creating different lifecycles on a per Action basis via Interceptor Stacks. Custom stacks can be created and used with different Actions, as needed.
- In Struts2 doesn't have thread-safety issues as Action objects are instantiated for each request. A servlet container generates many throw-away objects per request, and one more object does not impose a performance penalty or impact garbage collection.
- Struts2 Actions can be tested by instantiating the Action, setting properties, and invoking methods. Dependency Injection support also makes testing simpler.
- Struts2 can use JSTL, but it also supports a more powerful and flexible expression language called "Object Graph Notation Language" (OGNL).
- Struts2 uses OGNL for type conversion and converters to convert Basic and common object types and primitives as well.
- Struts2 allows manual validation that is done by using the validate method and the Xwork Validation framework. The Xwork Validation Framework allows chaining of validations into

sub-properties using the validations defined for the properties class type and the validation context

- Struts2 uses a ValueStack technology to make the values accessible to the taglibs without coupling the view to the object to which it is rendering. The ValueStack strategy enables us to reuse views across a range of types, having same property name but different property types.

Struts2 architecture

Let's try to understand the standard architecture of Struts2 application by this simple Figure 1.3:

- Step – 1** User sends a request for the action
- Step – 2** Container maps the request in the Web.xml file and gets the class name of controller.
- Step – 3** Container invokes the controller (StrutsPrepareAndExecuteFilter or FilterDispatcher). Since Struts2.1, it is StrutsPrepareAndExecuteFilter. Before 2.1 it was FilterDispatcher.
- Step – 4** Controller gets the information for the action from the ActionMapper
- Step – 5** Controller invokes the ActionProxy
- Step – 6** ActionProxy gets the information of action and interceptor stack from the configuration manager which gets the information from the Struts.xml file.
- Step – 7** ActionProxy forwards the request to the ActionInvocation
- Step – 8** ActionInvocation invokes each interceptors and action
- Step – 9** A result is generated
- Step – 10** The result is sent back to the ActionInvocation
- Step – 11** A HttpServletResponse is generated
- Step – 12** Response is sent to the user

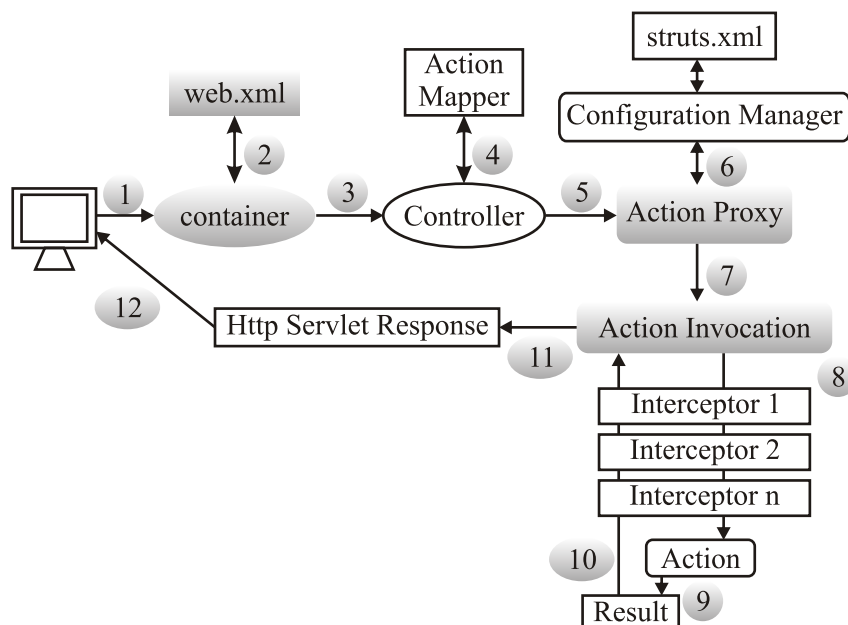


Figure 3 : Architecture of struts 2.

Related Work

Web and Internet is ever growing area and the demands for the applications are growing. A single framework is not capable to handle the architecture of the application. To meet the current requirements of the applications it's necessary to design architecture to implement the Frameworks.

Struts Framework have been designed and developed for the front end control of the Web applications. It provides the various features for the applications that interact to the users. It also follows the MVC2 design features. Spring Framework is designed to handle the various tasks. The Spring work for the desktop and internet based applications also. It follows the principals of the MVC2.

The simultaneous use of the Struts and Spring frameworks in the single application with the applying the MVC Design principals so that we can improve the performance of the applications.

Struts Framework consists of three major blocks, which are described in Figure 4.

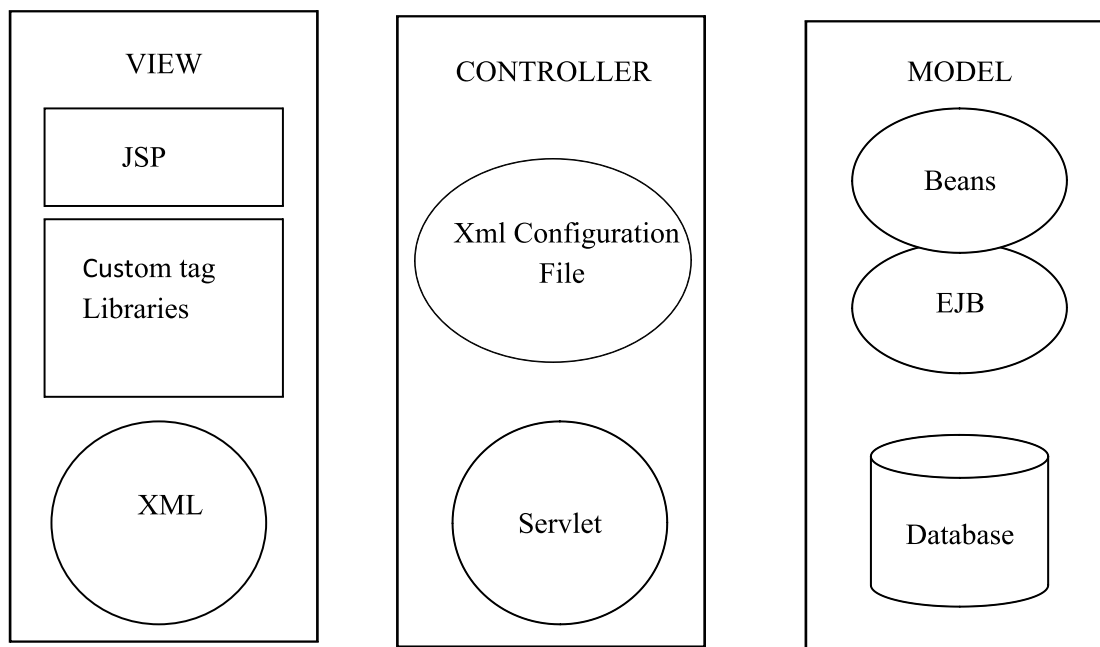


Figure 4: MVC2 Model.

First is The View Block which controls the presentation part of the complete model. This contains following JSP files which you write for your specific application set of JSP custom tag libraries Resource files for Internationalization.

Second Block is representing the Controller. This is for navigation the complete application. This contains XML configuration files; it contains the tags for the navigation of the paths.

Third Block is the Model. This part do the work of the Business Logic, Fetching and storing data to the database. This contains following Java Beans Enterprise Java Beans Database. Following figure shows the working of the components in the Struts framework (Gupta *et al.*, 2010).

In this section a computer components distribution system has been designed using the architecture proposed in the foregoing sections. The business features include user authentication, online product catalogs, shopping cart, special functions, order generating, checkout features and Email confirmation. According to system requirements the following Servlet and JSP documents are defined: login page, home page, product catalog page, the product details page, View Cart page, order confirmation page. The

framework is shown in Figure 1.4. The MVC pattern is adopted in Figure 1.4, that is, business logic tier represents the Model, JSP documents are the View, and Servlet serves as the Controller. There're no direct calls among JSP documents. The Servlet receive all HTTP request, and call the appropriate business logic model, then run different JSP documents according to the processing results. End clients can receive HTML response which can be seen in a browser. In Figure.1.4 the steps of accessing the database are: firstly Servlet calls the Session Bean, then the Session Bean calls entity beans, finally the database is operated by the Entity Bean.

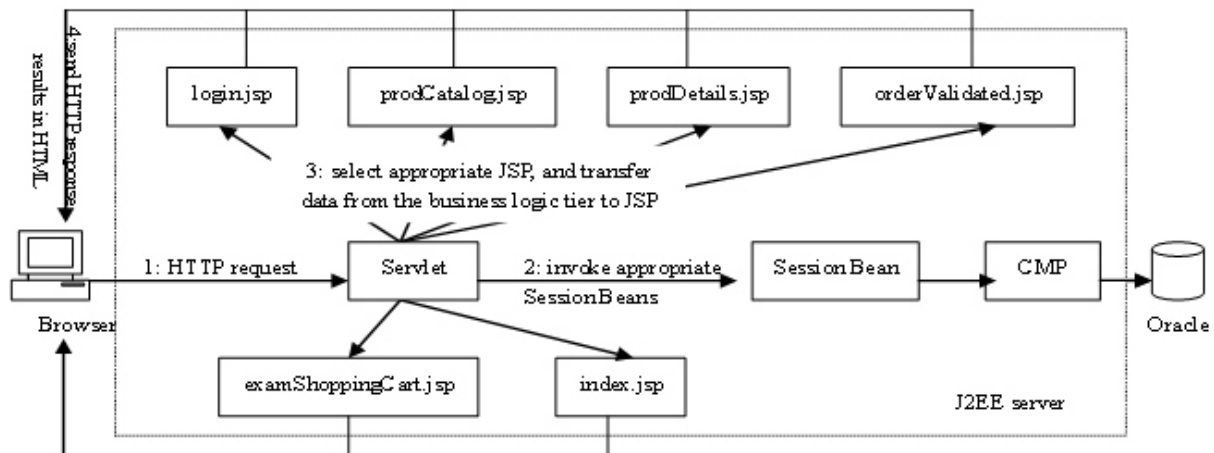


Figure 5: Architecture of Cart.

This approach not only separates the business logic from the data persistence logic and simplifies the development process, but also develops a stronger e-Commerce system with scalability and maintainability.

Implementation

In this part of paper we have discuss ONGL implementation of Struts2 with Tomcat (open source servlet container developed by the Apache Software Foundation (ASF)) (Saxean, 2011). Tomcat implements the Java Servlet and the Java Server Pages (JSP) specifications from Sun Microsystems, and provides a "pure Java" HTTP web server environment for Java code to run (Saxean, 2011).

ONGL is the Object Graph Navigation Language which is an open source framework from Apache Struts2. It is used in setting and getting properties from the Java beans. OGNL also enables to invoke the methods from the Java classes. OGNL is primarily designed to use with the UI forms to set the form values to the JavaBeans .OGNL was integrated of Strts2.

The Object Graph Navigation Language is an Expression language. It simplifies the accessibility of data stored in the ActionContext.

The Struts2 sets the ValueStack as the root object of OGNL. Notice that action object is pushed into ValueStack .we can direct access the action property.

The ValueStack strategy enables us to recuse views across a range of types, having same property name but different types.

```
<s:property value="name">
```

Here name is the property key. The Struts framework placed other objects in ActionContext also e.g. map representing the request. Session, application scopes.

To get these values i.e. not the action property,we need to use # notation. For example to get the data from

session scope , we need to use #session as given below

```
<s:property name="#.session.name"/>
```

or

```
<s:property name="#session.[name]"/>
```

ONGL and ValueStack work together in Struts2 to handle a request. ONGL offers the expression language which can be a part form tags and UI pages. The expression language helps to map the entered values to the destination properties of the Java Beans to which they have to be set.

When a form is submitted, the values are transmitted as String through the HTTP Protocol. These String values have to be converted to the respective primitive or custom Java types to set them in the beans. During the process of setting the values, ONGL consults the type converters available in Strus2 and helps in conversions of the form values to the destination types and will set into the JavaBeans.

Finally ONGL helps in three areas in Struts2 during the request handling Process

- ONGL offers the expression language which can be used in the forms and response pages.
- OGNL helps in the type conversions during setting the form values to the Java Beans types.
- Helps in looking for objects in ValueStack.

In this application you will learn about ONGL with software project. First let's see how to access an array of String variable using OGNL.in the action classs we create a string array and initialize shown below:

```
package ankur;
public class SampleArray
{
    private String [ ] Arr;
    {
        Arr= new String [ ] {"Java","J2ee","Strus"};
    }

    public String execute()
    {
        return "success";
    }

    public String [ ] getArr()
    {
        return Arr;
    }

    public void setArr(String [ ] Arr)
    {
        this.Arr=Arr;
    }
}
```

You can access the array values in the jsp page using the OGNL expression in the following way

```
<b> Array Example  
<br><hr>  
<b> sampleArray:</b><s:property value="Arr"/><br>  
<b> sampleArray.length:</b><s:property value="Arr.length"/><br>  
<b> sampleArray[0]:</b><s:property value="Arr[0]"/><br>  
<b> [0].sampleArray:</b><s:property value="[0].Arr"/><br>  
<b> top.sampleArray:</b><s:property value="top.Arr"/><br>
```

Since our object is on top of the ValueStack we can access it using [0] notation .if we need second position from the top, we will access it using (Wang et al. 2009)notation.

Now let's see how to access an ArrayList using ONGL expression language . in the action class we create and initialize the ArrayList is shown below:

```
package ankur;  
import Java.util.*;  
public class SampleArrayList  
{  
private List<String> List1=new ArrayList<String>();  
List1.add("Java");  
List1.add("J2ee");  
List1.add("Strus");  
}  
public String execute()  
{  
return "success";  
}  
public List<String> getList1()  
{  
return List1;  
}  
public void setList1(List<String> List1)  
{  
this.List1=List1;  
}  
}
```


You can access the ArrayList values in the jsp page using the OGNL expression in the following way:

```
<b> ArrayList Example  
<br><hr>  
<b> sampleArrayList:</b><s:property value="List1"/><br>  
<b> sampleArrayList.size:</b><s:property value="List1.size"/><br>  
<b> sampleArrayList[0]:</b><s:property value="List1[0]"/><br>
```

Now let's see how to access the name property of the User Object in the action class using the OGNL expression language .the code has given below:

```
package ankur;  
public class DemoAction  
{  
private User user =new User();  
{  
User.setName("Ankur");  
}  
public String execute()  
{  
return "success"  
}  
public String getQuote()  
{  
return "welcome";  
}  
public User getUser()  
{  
return user;  
}  
  
public User setUser(User user)  
{  
this.user=user;  
}  
}
```

You need to use second-level ONGL expression language to access the user name property.

```
<b> user.name:</b> <s:property value="user.name" /><br>
<B>quote():</b><s:property value="quote()"/><br>
```

Future work

Open source technologies are the best to attract the academic and research scholar to work. Struts framework is a best implementation of MVC, POJO based architecture. Struts are a powerful technology for business data, and it enables Application to access logic from any source in a platform-independent manner. Future work of this report is to develop an enterprise application which is based on Struts, Spring and Hibernate.

Conclusion

This research has proposed to solve the problem of the data expression, based on the development architecture framework composed by the Struts. It can protect the business data effectively. Besides, it is useful to the upper debug and maintenance by separating the business logic and the data expression, and also by separating the data in different layers. With the development of the Web, it is absolutely necessary for a large scale enterprise to be informational. The Struts emphasize particularly on the control of the business logic. It will also open the new scope and new business opportunities' for the companies and the programmers. It's recommended to use this Framework for the Better performance.

References

- Brown, D., Davis, M.C., Stanlick, S. 2011. Struts in Action, *DreamTech*, Delhi, India, 10
- Dey, T. 2011. A Comparative Analysis on Modeling and Implementing with MVC Architecture. *International Conference on Web Services Computing (ICWSC) Proceedings published by International Journal of Computer Applications® (IJCA)* 44.
- Garg, R., Sood, Y., Kottana, B., Totlani, P. 2011. A Framework Based Approach for the Development of Web Based Applications. *World of Computer Science and Information Technology Journal (WCSIT)*, 1(1), 1-4.
- Gupta, P., Govil, M. 2010. Spring Web MVC Framework for rapid open source J2EE application development: a case study, *International Journal of Engineering Science and Technology*. 2(6), 1684-1689.
- Gupta, P., Govil, M.C. 2010. MVC Design Pattern for the multi framework distributed applications using XML, spring and Struts framework. *International Journal on Computer Science and Engineering*, 2(4), 1047-1051.
- Saxean, A. 2012. Java Server Pages *MJP*. Chennai. Tamilnadu. india. 341, 342.
- Saxean, A., Jakhmola, R. 2011. Securing Confidential Data using Java/J2EE, *IJSTM*, Vol. 2 Issue 3.
- Saxena, A. 2011. Handling of Synchronized Data using Java/J2ee, *International Journal of Management. IT & Engineering Volume 1*, Issue 7.
- Wang, Y., Guo, C., Song, L. 2009. Architecture of E-Commerce Systems Based on J2EE and MVC Pattern, *IEEE Computer Society*.
- Wojciechowski, J., Sakowicz, B., Dura, K., Napieralski, A. 2004. MVC Model, Struts Framework and File upload Issues in Web Applications Based on J2EE Platform, *TCSET*.