Assignment No: 08

* **Title :-**

   Design & develop any web application using struts framework.

* **Objective :-**

   1) To import the efficient & available client side & serverside technologies.

   2) To implement the communication between computing nodes using client side & server side.

* **Problem Statement:**

   Create a login module for the web application using struts framework.

* **Outcomes :-**

   1) Implement the effective client server tech-nologies using struts framework.

   2) Solve the complex problem of development using MVC framework.

* **Software & Hardware Requirement**

   software Java 17 or Higher, Apache Tomcat or higher.

* **Theory :-**

The framework plays a vital role in industries for manageable & well designed application development as well as enterprise application development. The core of the Struts framework is flexible control layer based on standard techno technologies like java servlet Java Beans Resource bundles & VMC as well as various Jakarta commons packages. Struts encour-ages application architecture based on the model.

* Model - View Controller Architecture:
  -Model-view-controller is a way to avoid build application that promotes complete seperation between business logic & presentation. It is not specifies to web application. or Java or J2EE the view is the user inter-face, the Screen that the end user interface, the Screen that the end user of the application actually see & interacts with in a J2EE web application, views are JSP files.

* what is struts :-

Struts is a framework that advances the utilization of the Model - View - Controller engineering for planning substantial scal application. The structure incorporates an arrangement of custom label libraries & their related Java classes alongside different utility classes.

* Struts Tags : Common Attributes :
Almost all tags provided by the Struts framework use the foll" attributes id, name, property scope.

* Creating HTML Tags :-
Frequently data should be gather form a client & handled without the capacity together client input, a web application would be futile, so as to get client data, a html shape is utilized client inform- ation can originate from a few gadgets. For example, content fields content boxes, check takes care of pop means & radio catches.

* Conclusion :~
Hence, we have successfully tested the struts framework & tested the results.

Name:

Fokane Sakshi A

Roll No: 42

Div: TE-A

Assignment 8 : Create a login module for the web application using struts framework.

LoginForm.java

```java
package com.example.javawebtutor.form;

import javax.servlet.http.HttpServletRequest;

import org.apache.struts.action.ActionForm;
import org.apache.struts.action.ActionMapping;

public class LoginForm extends ActionForm {

    private String userName = null;
    private String password = null;

    public String getUserName() {
        return userName;
    }

    public void setUserName(String userName) {
        this.userName = userName;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }
```

```java
    @Override
    public void reset(ActionMapping mapping, HttpServletRequest request) {
        this.password = null;
    }
```
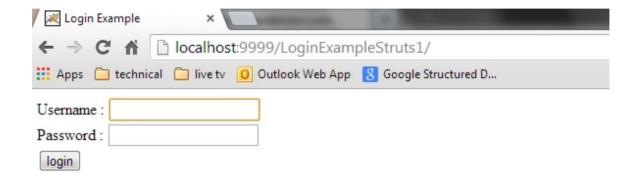
LoginAction.java

```java
package com.example.javawebtutor.action;

import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.apache.struts.action.Action; import
org.apache.struts.action.ActionForm; import
org.apache.struts.action.ActionForward; import
org.apache.struts.action.ActionMapping;

import com.example.javawebtutor.form.LoginForm;

public class LoginAction extends Action
    { @Override
    public ActionForward execute(ActionMapping mapping, ActionForm
        form, HttpServletRequest request, HttpServletResponse response)
        throws Exception {
    LoginForm loginForm = (LoginForm) form;

    if (loginForm.getUserName() == null || loginForm.getPassword() == null
        |    !loginForm.getUserName().equalsIgnoreCase("Mukesh")
        |    !loginForm.getPassword().equals("kumar")) {
        return
    mapping.findForward("success"); } else
        return mapping.findForward("failure");
    }

}
```

Struts-config.xml

```xml
<?xml version="1.0" encoding="UTF-8"?
> <!DOCTYPE struts-config PUBLIC
"-//Apache Software Foundation//DTD Struts Configuration
1.3//EN" "http://struts.apache.org/dtds/struts-config_1_3.dtd">

<struts-config>
   <form-beans>
      <form-bean name="loginForm"
type="com.example.javawebtutor.form.LoginForm"/>
   </form-beans>
   <action-mappings>
      <action name="loginForm" path="/login"
type="com.example.javawebtutor.action.LoginAction"
scope="request" input="/login.jsp">
         <forward name="failure" path="/login.jsp" redirect="true"/>
         <forward name="success" path="/success.jsp" redirect="true"/>
      </action>
   </action-mappings>
</struts-config>
```

Login.jsp

```jsp
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
pageEncoding="ISO-8859-1"%>
<%@ taglib uri="http://struts.apache.org/tags-html" prefix="html"%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd"> <html>

<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<title>Login Example</title>
</head>
<body>
   <html:form action="/login" focus="userName">
```

```
Username : <html:text property="userName" />
    <br>
Password : <html:password property="password" /
    > <br>
    <html:submit value="login" />
</html:form>

</body>
</html>
```

Success.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"

    pageEncoding="ISO-8859-1"%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01
Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">

<title>Successful Login Page</title>

</head>

<body>

    <h2>Welcome Boss</h2>

</body>

</html>
```

Index.jsp

```
<jsp:forward page="/login.jsp"></jsp:forward>
```

Welcome Boss