K J's Educational Institute

K J College of Engineering & Management Research

(Accredited by NAAC)

Sr. No. 25 & 27, Kondhwa - Saswad Road, Near Bopdev Ghat, Pune 411048.

Department of Computer Engineering

Savitribai Phule Pune University
Third Year of Computer Engineering (2019 Course)
310257: Web Technology Laboratory

Teaching Scheme **Practical:** 02 Hours/Week	**Credit:**01	**Examination Scheme and Marks Term work:** 25 Marks **Oral:** 25 Marks

Companion Course: Web Technology (310252)

Subject Incharge	Prof. Geetanjali D. Bansod
Academic Year	2023-24
Semester	II

Course Objectives:

- To learn the web based development environment
- To use client side and server side web technologies
- To design and develop web applications using front end technologies and backend databases

Course Outcomes:

On completion of the course, learners will be able to

CO1: Understand the importance of website planning and website design issues

CO2: Apply the client side and server side technologies for web application development

CO3: Analyze the web technology languages, frameworks and services

CO4: Create three tier web based applications

K J's Educational Institute

K J College of Engineering & Management Research

(Accredited by NAAC)

Sr. No. 25 & 27, Kondhwa - Saswad Road, Near Bopdev Ghat, Pune 411048.

Department of Computer Engineering

List of Experiments

Contents

7	Build a dynamic web application using PHP and MySQL. a. Create database tables in MySQL and create connection with PHP. b. Create the add, update, delete and retrieve functions in the PHP web app interactingwith MySQL database	29
8	Design a login page with entries for name, mobile number email id and login button. Use struts and perform following validations Validation for correct names Validation for mobile numbers Validation for email id Validation if no entered any value Re-display for wrongly entered values with message Congratulations and welcome page upon successful entries	34
9	Design an application using Angular JS. e.g., Design registration (first name, last name, username, password) and login page using Angular JS.	37
10	Design and implement a business interface with necessary business logic for any web application using EJB. e.g., Design and implement the web application logic for deposit and withdraw amount transactions using EJB.	43
11	Mini Project: Design and implement a dynamic web application for any business functionality by using web development technologies that you have learnt in the above given assignments.	

Experiment No. 1

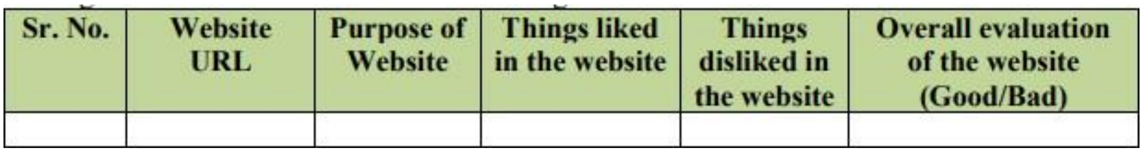
TITLE: Case Study

OBJECTIVES:

Understand about different Website design issues.

PROBLEM STATEMENT

Case Study: Before coding of the website, planning is important, students should visit different websites (Min. 5) for the different client projects and note down the evaluation results for these websites, either good website or bad website in following format:



Sr. No.	Website URL	Purpose of Website	Things liked in the website	Things disliked in the website	Overall evaluation of the website (Good/Bad)

From the evaluation, students should learn and conclude different website design issues, which should be considered while developing a website

OUTCOME

Students will be able to,

1. Find out different Web Site Design Issues.
2. How to design a good website?

SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements: Browser

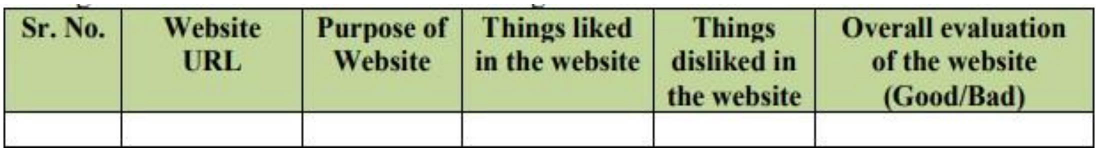
THEORY-CONCEPT

Parameters to check a Website Quality:

1. Site Purpose
2. Target Audience
3. Responsive/ Mobile Friendly
4. Fresh Content or Outdated Content
5. Using understandable language on the web pages
6. Visual Design/ Quality of Images
7. Look and feel (Using a uniform look and feel)
8. Engaging Content (Making the site interesting)
9. Navigation (Making the site easy to use)
10. Slow upload speed
11. Contents Representation (Font Size/Style)

DESIGN / EXECUTION STEPS

Before coding of the website, planning is important, visit different websites (Min. 5) for the different client projects and note down the evaluation results for these websites, either good website or bad website in following format by considering above parameters.



Sr. No.	Website URL	Purpose of Website	Things liked in the website	Things disliked in the website	Overall evaluation of the website (Good/Bad)

From the evaluation, conclude different website design issues, which should be considered while developing a website

CONCLUSION/ANALYSIS

Hence, Students has learned what are different design issues and how to design good website.

ORAL QUESTIONS

1. What are different parameters to find good quality of website?
2. What are different web design issues?
3. What are advantages of websites?
4. What are disadvantages of websites?
5. If you are told to design website which website, you will design? Why?

Experiment No. 2

TITLE

Title: HtML, CSS

OBJECTIVES

1. Understand about basic concepts of html
2. Understand the basic concepts of CSS

PROBLEM STATEMENT

Implement a web page index.htm for any client website (e.g., a restaurant website project) using following: a. HTML syntax: heading tags, basic tags and attributes, frames, tables, images, lists, links for text and images, forms etc. b. Use of Internal CSS, Inline CSS, External CSS

OUTCOME

Students will be able to,

1. Design static webpage using HTML.
2. Apply CSS to HTML pages.

SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements:

1. Operating System: Windows 7/8/10/Ubuntu
2. Browser: Firefox/Google Chrome/ Microsoft Edge etc.
3. Software: Sublime Text Editor/Notepad/ Notepad ++

Hardware Requirements:

1. Processor: Minimum 1 GHz.
2. Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
3. Hard Drive: Minimum 32 GB.
4. Memory (RAM): Minimum 1 GB
5. Sound card-speakers/camera/microphone (Depending upon website selection)

THEORY-CONCEPT

- HTML: HTML is the standard markup language for creating Web pages.
- HTML stands for Hyper Text Markup Language
- HTML describes the structure of Web pages using markup
- HTML elements are the building blocks of HTML pages

- HTML elements are represented by tags
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on
- Browsers do not display the HTML tags, but use them to render the content of the page

□ HTML Versions:

Table.1: HTML Versions

HTML	1991
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML 5	2014

CSS:

CSS stands for Cascading Style Sheet. It is nothing, but design language intended to simplify the process of making web pages presentable.CSS handles the feel and look part of a web page. By using CSS, one can control the color of text, style of fonts, spacing between paragraphs, layout designs.

CSS is easy to learn, easy to understand and it provides powerful control on presentation of an HTML document.

Advantages of CSS:

It saves the time, Pages load faster, Easy maintenance, Superior styles to HTML, Multiple Device Compatibility, Global web standards, Offline Browsing, Platform Independence.

CSS3 Modules:

CSS3 Modules are having old CSS specifications as well as extension features.

- Box Model
- Selectors
- Background
- Border
- Image Values and Replaced Content
- Text Effects
- Animations
- 2D/3D Transformations

- Multiple Column Layout
- User Interface

TECHNOLOGY / TOOL

- The <!DOCTYPE html> declaration defines this document to be HTML5
- The <html> element is the root element of an HTML page
- The <head> element contains meta information about the document
- The <title> element specifies a title for the document
- The <body> element contains the visible page content

- The <h1> element defines a large heading
- The <p> element defines a paragraph
- HTML tags are element names surrounded by angle brackets:

```
<tagname>content goes here...</tagname>
```

CSS can be added to HTML elements in 3 ways:

- **Inline** - by using the style attribute in HTML elements. An inline CSS is used to apply a unique style to a single HTML element.

  Ex. <h1 style="color:blue;">This is a Blue Heading</h1>

- **Internal -** by using a <style> element in the <head> section. An internal CSS is used to define a style for a single HTML page. An internal CSS is defined in the <head> section of an HTML page, within a <style> element.

Example:

```
<style>

body {background-color: yellow ;}

h1 {color: blue;}

p   {color: red;}

</style>
```

- **External** - by using an external CSS file. An external style sheet is used to define the style for many HTML pages. With an external style sheet, you can change the look of an entire web site, by changing one file!To use an external

style sheet, add a link to it in the <head> section of the HTML page.
Example:

```
<link rel="stylesheet" href="styles.css">
```

- Use the HTML **<head>** element to store <style> and <link> elements
- Use the CSS **color** property for text colors
- Use the CSS **font-family** property for text fonts
- Use the CSS **font-size** property for text sizes
- Use the CSS **border** property for borders
- Use the CSS **padding** property for space inside the border
- Use the CSS **margin** property for space outside the border

DESIGN / EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Write the HTML code in notepad and save with .html extension.
2. Write the CSS code in notepad and save with .css extension.
3. Import CSS file in HTML page.
4. Open HTML page in the browser.

CONCLUSION/ANALYSIS

Hence, we have designed static web pages using HTML and CSS.

ORAL QUESTIONS

1. What is the difference between HTML and HTML5?
2. What is the difference between html elements and tags?
3. What is marquee?
4. What is the use of span tag? Give an example?
5. What is the use of ‘required ’attribute in HTML5?
6. What is External stylesheet? What are the advantages and disadvantages?
7. What is CSS selector?
8. What are the components of CSS style?
9. What is browser safe color?

Experiment No. 3

TITLE

Title: XML, DTD and CSS/XSL.

OBJECTIVES

PROBLEM STATEMENT

Design the XML document to store the information of the employees of any business organization and demonstrate the use of:

a) DTD

b) XML Schema

And display the content in (e.g., tabular format) by using CSS/XSL

OUTCOMES

Students will be able to,

1. Design static webpage using XML.
2. Apply DTD to XML pages.
3. Apply CSS/XSLT to XML pages

SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements:

1. Operating System: Windows 7/8/10/Ubuntu
2. Browser: Firefox/Google Chrome/ Microsoft Edge etc.
3. Software: Sublime Text editor/ Notepad/ Notepad++, Eclipse (for DTD)

Hardware Requirements:

1. Processor: Minimum 1 GHz.
2. Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
3. Hard Drive: Minimum 32 GB.
4. Memory (RAM): Minimum 1 GB
5. Sound card-speakers/camera/microphone (Depending upon website selection)

THEORY-CONCEPT

XML stands for Extensible Markup Language. It is nothing but the text-based markup language which is derived from Standard Generalized Markup Language (SGML). XML tags identify the data and are used to store and organize the data, rather than specifying how to display it like HTML tags, which are used to display the data. XML is not going to replace HTML in the near future, but it introduces new possibilities by adopting many successful features of HTML.

There are three important characteristics of XML that make it useful in a variety of systems and Solutions:

1. XML is extensible – XML allows you to create your own self-descriptive tags, or language, that suits your application.
2. XML carries the data, does not present it – XML allows you to store the data irrespective of how it will be presented.
3. XML is a public standard – XML was developed by an organization called the World Wide Web Consortium (W3C) and is available as an open standard.

TECHNOLOGY/TOOL

The XML document have an XML declaration, but it is optional, and it is written as–

<? xml version = "1.0" encoding = "UTF-8"?>

Where version is nothing but the version of an XML document and UTF specifies the character-encoding used in the document. Each XML-element needs to be closed either with start or with end elements as shown below –

<element>………</element>

An XML document can have only one root element.

XML Attributes:

Using a name/value pair, an attribute specifies a single property for an element. An XML-element can have one or more attributes. For example –

<a href = "http://www.google.com/">XMLTutorial</a>

Here href is the attribute name and http://www.google.com/ is attribute value.

DTD

A DTD is a Document Type Definition. A DTD defines the structure and the legal elements and attributes of an XML document. With a DTD, independent groups of people can agree on a standard DTD for interchanging data. An application can use a DTD to verify that XML data is valid.

An Internal DTD Declaration

If the DTD is declared inside the XML file, it must be wrapped inside the <!DOCTYPE> definition:

XML document with an internal DTD

```
    <?xml version="1.0"?>
    <!DOCTYPE note [
    <!ELEMENT note (to,from,heading,body)>
    <!ELEMENT to (#PCDATA)>
    <!ELEMENT from (#PCDATA)>
    <!ELEMENT heading (#PCDATA)>
    ]>
<note>
<to>Neha</to>
<from>Amit</from>
<heading>Reminder</heading>
</note>
```

The DTD above is interpreted like this:

!DOCTYPE note defines that the root element of this document is note

!ELEMENT note defines that the note element must contain four elements: "to,from, heading"

!ELEMENT to defines the to element to be of type "#PCDATA"

!ELEMENT from defines the from element to be of type "#PCDATA"

!ELEMENT heading defines the heading element to be of type "#PCDATA"

An External DTD Declaration

If the DTD is declared in an external file, the <!DOCTYPE> definition must contain a reference to the DTD file:

```
<?xml version="1.0"?>
<!DOCTYPE note SYSTEM "note.dtd">
<note>
```

```
   <to>Tove</to>
   <from>Jani</from>
  <heading>Reminder</heading>
</note>
```

XSLT

XSL (eXtensible Stylesheet Language) is a styling language for XML.

XSLT stands for XSL Transformations.

The root element that declares the document to be an XSL style sheet is <xsl:stylesheet> or <xsl:transform>.

Note: <xsl:stylesheet> and <xsl:transform> are completely synonymous and either can be used!

The correct way to declare an XSL style sheet according to the W3C XSLT Recommendation is:

```
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">
```

Link the XSL Style Sheet to the XML Document

Add the XSL style sheet reference to your XML document ("student.xml"):

```
<?xml version="1.0" ?>
<?xml-stylesheet type="text/xsl" href="stud.xsl"?>
```

If you have an XSLT compliant browser it will nicely **transform** your XML into XHTML.

DESIGN/EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Write the XML code in notepad and save with .xml extension.
2. Write the DTD in notepad and save with .dtd extension (For external DTD)
3. Write a XSLT code in notepad and save using .xsl extension.
4. Open XML page in the browser for running simple XML or XML using XSLT.
5. To run internal/external dtd open Eclipse and run the code using validate.

TEST CASES

Manual testing is used to check whether XSLT gets applied or not.

Eclipse validates function used to check whether DTD gets applied or not.

CONCLUSION/ANALYSIS

Hence, we have designed static web pages using XML, XSLT/CSS and DTD

ORAL QUESTIONS

1. Explain difference between HTML and XML?
2. What is XML DOM?
3. Explain difference between CDATA and PCDATA?
4. What is mean by simple element and complex element?
5. What is DTD?
6. Explain XSL and XSL

Experiment No. 4

TITLE: HTML, Java Script

OBJECTIVES

1. Understand the basic concepts of JavaScript.
2. Use JavaScript for validation of data.

PROBLEM STATEMENT

Implement an application in Java Script using following:

a) Design UI of application using HTML, CSS etc.

b) Include Javascript validation

c) Use of prompt and alert window using Javascript

e.g., Design and implement a simple calculator using Javascript for operations like addition, Multiplication, subtraction, division, square of number etc.

a) Design calculator interface like text field for input and output, buttons for numbers and Operators etc.

b) Validate input values

c) Prompt/alerts for invalid values etc

OUTCOMES

Students will be able to,

1. Design a static webpage using HTML.
2. Apply JavaScript to HTML pages for taking inputs from users and validation of inputs and perform basic operations.

SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements:

1. Operating System: Windows 7/8/10/Ubuntu
2. Browser: Firefox/Google Chrome/ Microsoft Edge etc.
3. Software: Sublime Text editor/ Notepad/ Notepad++, Eclipse (for DTD)

Hardware Requirements:

1. Processor: Minimum 1 GHz.
2. Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)

3. Hard Drive: Minimum 32 GB.
4. Memory (RAM): Minimum 1 GB
5. Sound card-speakers/camera/microphone (Depending upon website selection)

THEORY-CONCEPT

JavaScript is a programming language of HTML as well as web. It is preferred for creating network-centric applications. It is integrated and complementary

with Java. As JavaScript is integrated with HTML it is very easy to implement. It is open as well as cross-platform.

Advantages:

The advantages of using JavaScript are –

- It requires less server interaction
- Immediate feedback to the visitors
- Increased interactivity
- Richer interfaces

Validation:

When a client enters all the necessary data and presses the submit button, form validation is done at server side If data entered by a client is incorrect or missing, the server needs to send all data back to the client and request for resubmission of form with correct information. This is really a lengthy process which puts a lot of load(burden) on the server.

So, JavaScript provides a way to validate form's data on the client's side itself before sending it to the web server. Form validation performs two functions-

□ Data Format Validation – Secondly, the data that is entered must be checked for correct

Format and its value. The code must include appropriate logic to test correctness of data.

TECHNOLOGY/TOOL

JavaScript can be implemented using JavaScript statements that are placed within the <script>.

You can place the <script> tags, containing your JavaScript, anywhere within your web page, but it is normally recommended that you should keep it within the <head> tags.

The script tag takes two important attributes:

- **Language** – This attribute specifies what scripting language you are using. Typically, its value will be JavaScript. Although recent versions of HTML (and XHTML, its successor) have phased out the use of this attribute.
- **Type** – This attribute is what is now recommended to indicate the scripting language in use and its value should be set to "text/JavaScript".

DESIGN/EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Write a code in notepad/Notepad++/Sublime editor and save with .html extension.
2. Design UI of application using HTML, CSS etc.
3. Include Javascript validation
4. Use of prompt and alert window using Javascript
   e.g., Design and implement a simple calculator using Javascript for operations like addition,

   Multiplication, subtraction, division, square of number etc.
5. Design calculator interface like text field for input and output, buttons for numbers and
   Operators etc.
6. Validate input values
7. Prompt/alerts for invalid values etc
8. Open HTML page in the browser.

TEST CASES

Manual testing is used to check following validations

☐ Input should be numbers and operations only.
☐ If instead of numbers some other input-like text is filled by the user then code must validate the same and an alert message should be printed.

CONCLUSION/ANALYSIS

Hence, we validated the data using JavaScript.

ORAL QUESTIONS

1. Name some Javascript features.
2. How to define anonymous functions?
3. What is a callback?

4. What is the difference between undefined and not-defined in JavaScript?
5. What is ‘closure’ in JavaScript?
6. What are JavaScript data types?
7. What are all the types of Pop up boxes available in JavaScript?

Experiment No. 5

TITLE: Servlet

OBJECTIVE:

1. Understand about basic concepts of html, CSS

2. Understand the basic functionalities of Servlets

PROBLEM STATEMENT:

Implement the sample program demonstrating the use of Servlet. e.g., Create a database table ebookshop (book_id, book_title, book_author, book_price, quantity) using databases like Oracle/MySQL etc. and display (use SQL select query) the table content using servlet.

OUTCOME:

Students will be able to,

1. Develop a dynamic webpage using HTML and Servlet.

2. Write a server side java application called Servlet to catch the data sent from the client, process it and show the output.

SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements:

1. Operating System: Windows 7/8/10/Ubuntu

2. Browser: Firefox/Google Chrome/ Microsoft Edge etc.

3. Software/Editor : Eclipse

4. Any Operating System

5. JDK 7or later

6. Tomcat 7 or later

Hardware Requirements:

1. Processor: Minimum 1 GHz.

2. Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)

3. Hard Drive: Minimum 32 GB.

4. Memory (RAM): Minimum 1 GB

5. Sound card-speakers/camera/microphone (Depending upon website selection)

THEORY-CONCEPTS :

Servlet:

A Servlet is a server side program written in Java. Servlet is a web component that is deployed on the server for creating dynamic web pages. A Java servlet is a Java program that extends the capabilities of a server. Although servlets can respond to any types of requests, they most commonly execute applications hosted on Web servers.

Java Servlets are Java classes run by a web server that has an interpreter that supports the Java Servlet specification.

Servlets can be created using the packages

- javax.servlet
- javax.servlet.http

Servlet Life Cycle

- The servlet is initialized by calling the init() method.
- The servlet calls service() method to process a client's request.
- The servlet is terminated by calling the destroy() method.
- Finally, servlet is garbage collected by the garbage collector of the JVM.

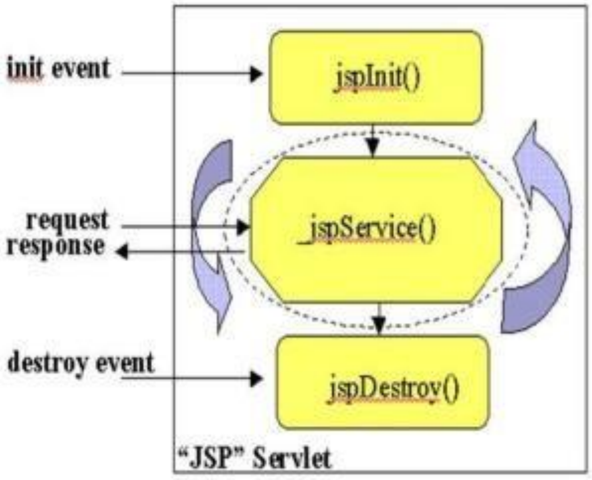


Figure: Functions of Servlet

TECHNOLOGY/TOOL

How to configure tomcat server in Eclipse? (One time Requirement)

- If you are using Eclipse IDE first time, you need to configure the tomcat server First.
- For configuring the tomcat server in eclipse IDE,
- click on servers tab at the bottom side of the IDE -> right click on blank area -> New -> Servers -> choose tomcat then its version -> next -> click on Browse button -> select the apache tomcat root folder previous to bin -> next -> addAll -> Finish.

Steps to run servlet in Eclipse

- Create a Dynamic web project
- create a servlet
- add servlet-api.jar file
- Run the servlet

Methods

- doGet-

A GET request results from a normal request for a URL or from an HTML form that has no METHOD specified and it should be handled by the doGet() method.

- doPost-

A POST request results from an HTML form that specifically lists POST as the METHOD and it should be handled by doPost() method

Reading HTML Form Data using Servlet

- getParameter() − You call the request.getParameter() method to get the value of a form parameter.
- getParameterValues() − Call this method if the parameter appears more than once and returns multiple values, for example checkbox.
- getParameterNames() − Call this method if you want a complete list of all parameters in the current request.

DESIGN /EXECUTION STEPS

Following steps are used to Create and Execute web applications,

1. Design html and servlet files with an extension of.html and .java

2. Start the Tomcat Server with port number

TEST CASES

.All records from the MySQL database should be displayed on the browser using JSP and HTML.

CONCLUSION / ANALYSIS

Hence, we have performed the dynamic web application using Servlet and MySQL.

ORAL QUESTIONS

1. What is Servlet?

2. What is the life-cycle of a servlet?

3. What is the difference between the Get and Post method?

4. What is the difference between GenericServlet and HttpServlet?

5. What is session tracking?

6. What are Cookies?

Experiment No. – 6

TITLE:

JSP- Java Server Pages

OBJECTIVES

1. Understand about basic concepts of html, CSS
2. Understand the basic functionalities of JSP
3. Having the knowledge of SQL query to create the database

PROBLEM STATEMENT

Implement the program demonstrating the use of JSP. e.g., Create a database table students_info (stud_id, stud_name, class, division, city) using database like Oracle/MySQL etc. and display (use SQL select query) the table content using JSP

OUTCOMES

Students will be able to,

1. Develop a dynamic webpage using JSP, HTML .
2. Write a server side application called JSP to catch the data sent from the client, process it and store it on a database (MySQL).

SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements:

Operating System: Windows 7/8/10/Ubuntu
Browser: Firefox/Google Chrome/ Microsoft Edge etc.
Software/Editor : Eclipse
Any Operating System
JDK 7or later
Tomcat 7 or later
Java MySQL Connector

Hardware Requirements:

1. Processor: Minimum 1 GHz.
2. Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)

3. Hard Drive: Minimum 32 GB.
4. Memory (RAM): Minimum 1 GB
5. Sound card-speakers/camera/microphone (Depending upon website selection)

THEORY-CONCEPT

Java Server Pages (JSP): It is a server side programming technology that is used to create dynamic web-based applications. JSP have right to use the complete Java APIs, including the JDBC API to access the databases.

It is a technology that helps software developers to create dynamic web pages based on HTML, XML and other document types. It was released in 1999 by Sun Microsystems. It is just like a PHP and ASP, but it uses the Java programming language.

A JSP element is a type of java servlet that is designed to accomplish the role of a user interface for a java web application. Web developers write JSPs as text files that combine HTML or XHTML code, XML elements, and rooted JSP actions and commands.

Using JSP, you can collect input from users through webpage forms, current records from a database or another source and create web pages dynamically.

JSP tags can be used for different purposes, such as retrieving information from a database or registering user preferences, accessing JavaBeans components, passing control between pages, and sharing information between requests, pages etc.

Why we need JSP?

JSP is used for the design of dynamic web page and servlet is used to code the logic that is present i.e. in the MVC (Model-View-Controller) architecture, the servlet is the controller and the JSP is the view.

Advantage of JSP over Servlet

1) Extension to Servlet

2) Easy to maintain

3) Fast Development: No need to recompile and redeploy

4) Less code than Servlet

Architecture of JSP

1. The request / response part of a JSP is defined in below architecture
2. The client initiated request for a JSP file using browser
3. Webs server (i.e, JSP Engine) invokes the JSP file and interpret the JSP file to produce a java code. The created java code will be a Servlet.
4. Once Servlet is created, the JSP engine compiles the servlet. Compilation errors will be detected in this phase.
5. Now the servlet class is loaded by the container and executes it.
6. Engine sends the response back to the client.

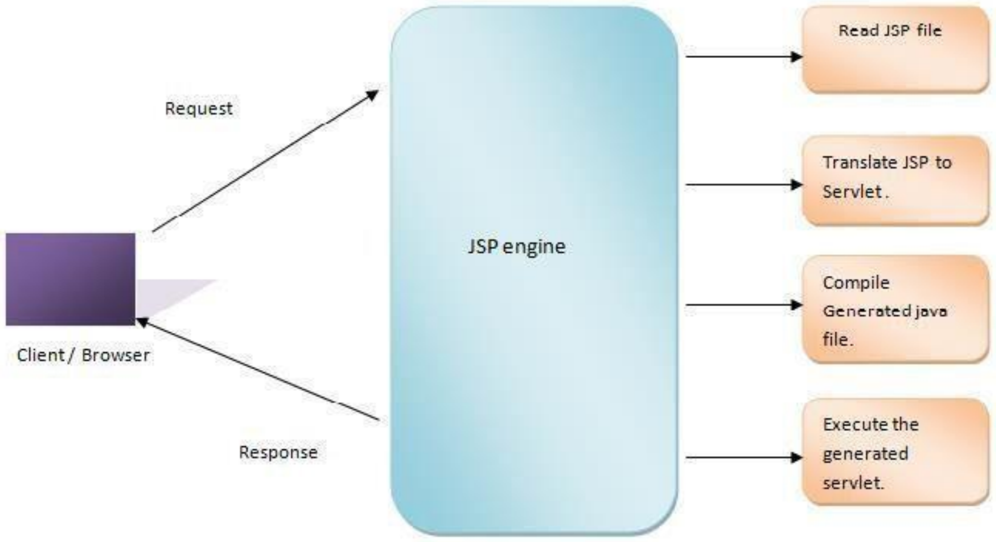

Request
Client / Browser
Response
JSP engine
Read JSP file
Translate JSP to Servlet.
Compile Generated java file.
Execute the generated servlet.

Figure. Architecture of JSP

Syntax of JSP:

JSP declaration is used to declare variables and methods as shown below,

<% text

Following is the simple and first example for JSP:

//Hello.jsp

<html>

<body>

<% out.println("Welcome to JSP Class"); %>

```
</body>

</html>
```

Output:

```
Welcome to JSP Class
```

Elements of JSP

Scripting Element	Example
Comment	<%-- comment --%>
Directive	<%@ directive %>
Declaration	<%! declarations %>
Scriptlet	<% scriplets %>
Expression	<%= expression %>

JDBC

Java JDBC is a java API to connect and execute query with the database. JDBC API uses jdbc drivers to connect with the database.

JDBC Driver is a software component that enables java application to interact with the database.

There are 4 types of JDBC drivers:

1. JDBC-ODBC bridge driver
2. Native-API driver (partially java driver)
3. Network Protocol driver (fully java driver)
4. Thin driver (fully java driver)

TECHNOLOGY/TOOL

1. JSP
2. IDE: Eclipse
3. Databases: MySQL

MySQL: MySQL is a freely available open source Relational Database Management System (RDBMS). It uses the Structured Query Language (SQL). SQL is the most popular language for adding, accessing and managing data in a database. It is most noted for its quick processing, proven reliability, ease and flexibility of use.

Steps to run JSP in Eclipse

DESIGN/EXECUTION STEPS

- In Eclipse-> File Menu-> Select New-> Dynamic web project
- Give Project Name
- You can find Project in Project Explorer window
- Right click on WebContent folder under Project ->New -> JSP File -> Give JSP file name with .jsp extension
- Type Code
- Right click on .jsp file name, Run AS-> Run on Server->Select Tomcat Server-> Finish

Important Note:

Copy MySQL Connector (Jar file) into Tomcat's Bin and Lib folder

- *Before running program right click on project name -> Build path -> Configure build path -> Libraries -> Add External Jar file -> Mysql .jar file -> ok*

TEST CASES

All records from the MySQL database should be displayed on the browser using JSP and HTML.

CONCLUSION/ANALYSIS

Hence, we have performed the dynamic web application using JSP and MySQL.

ORAL QUESTIONS

1. What is JSP?
2. What are the life-cycle methods for a JSP?
3. What are elements of JSP?
4. Difference between Servlet and JSP?
5. What is JDBC?

Experiment No. – 7

TITLE

Add dynamic web application essence using PHP, HTML and MySQL.

OBJECTIVES

1. To understand the principles and methodologies of PHP web based applications development process
2. Having the knowledge of SQL query to create the database

PROBLEM STATEMENT

Build a dynamic web application using PHP and MySQL.

a. Create database tables in MySQL and create connections with PHP.

b. Create the add, update, delete and retrieve functions in the PHP web app interacting with MySQL database.

OUTCOMES

Students should be able to,

Develop web based applications using suitable client side and server side web technologies.
Develop solutions to Database Related operations.

SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements:

Operating System: Windows 7/8/10/Ubuntu
Browser: Firefox/Google Chrome/ Microsoft Edge etc.
Software/Editor : Sublime Editor/Notepad/Notepad++
Any Operating System
XAMPP Server

Hardware Requirements:

1. Processor: Minimum 1 GHz.
2. Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
3. Hard Drive: Minimum 32 GB.
4. Memory (RAM): Minimum 1 GB
5. Sound card-speakers/camera/microphone (Depending upon website selection)

THEORY-CONCEPT

1. PHP:

The PHP Hypertext Preprocessor (PHP) began as a little open source venture that advanced as an ever increasing number of individuals discovered how valuable it was. Rasmus Lerdorf released the principal form of PHP route in 1994. PHP is a recursive acronym for "PHP: Hypertext Preprocessor".

PHP is a server side scripting dialect that is installed in HTML. It is utilized to oversee dynamic substance, databases, session following, even form whole internet business locales. It is incorporated with various prevalent databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.

PHP is pleasingly zippy in its execution, particularly when gathered as an Apache module on the Unix side. The MySQL server, once began, executes even extremely complex questions with colossal outcome sets in record-setting time.

PHP bolsters a substantial number of real conventions, for example, POP3, IMAP, and LDAP. PHP4 included help for Java and conveyed question designs (COM and CORBA), making n- level improvement a plausibility out of the blue. PHP is excusing: PHP dialect tries to be as pardoning as would be prudent. PHP Syntax is C-Like.

PHP performs framework capacities, i.e. from documents on a framework it can make, open, read, compose, and close them. PHP can deal with frames, i.e. accumulate information from records, spare information to a document; through email you can send information, return information to the client.

You include, erase, and adjust components inside your database through PHP. Access treats factors and set treats. Utilizing PHP, you can confine clients to get to a few pages of your site. It can encode information.

Example:

"Hello World" Script in PHP

To get a feel for PHP, first start with simple PHP scripts. Since "Hello, World!" is an essential example, first we will create a friendly little "Hello, World!" script. As mentioned earlier, PHP is embedded in HTML. That means that in amongst your normal HTML (or XHTML if you're cutting-edge) you'll have PHP statements like this –

```
<html>
<body>
<?php
 echo (“Hello Php”) ;
?>
</body></html>
```

To create and run PHP Web pages three fundamental parts should be introduced on your PC framework.

Web Server – PHP is server side scripting language so it requires Web server to execute(eg. Apache Tomcat)

Database – PHP will work with for all intents and purposes all database programming, including Oracle and Sybase yet most regularly utilized is uninhibitedly accessible MySQL database.

PHP Parser – keeping in mind the end goal to process PHP content directions a parser must be introduced to create HTML yield that can be sent to the Web Browser. This instructional exercise will manage you how to introduce PHP parser on your PC.

2. MySQL:

MySQL is the most famous Open Source Relational SQL Database Management System. MySQL is outstanding amongst other RDBMS being utilized for creating different online programming applications. MySQL is created, advertised and upheld by MySQL AB, which is a Swedish organization. This instructional exercise will give you a fast begin to MySQL and make you OK with MySQL programming.

TECHNOLOGY/TOOL

1. Technology is to be used is PHP (PHP Hypertext Preprocessor) and tool XAMPP server is to be used to execute PHP web application.
2. XAMPP server embeds the PHP, MySQL and phpmyadmin, these three tools must be required to run php web application.

For the design purpose html and CSS is to be used. For this design part contains the GUI of web applications, how its looks like? When users going to use the web application.

DESIGN/EXECUTION STEPS

Steps to install XAMPP and configure the PHP, MYSQL server.

1. Download the XAMPP using following link (download latest version as per your Operating system Windows/ Linux). https://www.apachefriends.org/download.html

2. Install XAMPP by running .exe file

3. Go to start->Xampp-> Xampp control panel. Start Apache and Mysql.

4. Goto Web browser(eg. Firefox) and write localhost and see Xampp has been started or not..

5. By Clicking on PhpMyAdmin you can create database, table and insert values in MySQL.

6. Open a note pad write a PHP code and save this file in Xampp->htdoc->create a folder here(eg.PHP1) and save file in this folder with name index.php

7. To run the code goto Firefox or any web browser and type in address bar - localhost/PHP1 your code will get run.

8. If file name is other that index.php then to run type in address bar – localhost/PHP1/hello.php (file name)

TEST CASES

All records from MySQL database should be displayed on browser using PHP and MYSQL.

CONCLUSION/ANALYSIS

In this assignment, we have studied how to design and develop small web application using PHP script, XAMPP server with apache server and MySQL as backend.

ORAL QUESTIONS

1. What is the use of "echo" in php?
2. How to include a file to a php page?
3. Differences between GET and POST methods?
4. What is the use of 'print' in php?
5. What is the difference between Session and Cookie?
6. What are the different errors in PHP?
7. How to print current date and time?

Experiment No: 8

TITLE

Title: Struts

OBJECTIVES

1. Understand the basic concepts of Struts.
2. Use Struts for Validation of user entered data.

PROBLEM STATEMENT

Design a login page with entries for name, mobile number, email id and login button.

Use struts and perform following validations

a. Validation for correct names

b. Validation for mobile numbers

c. Validation for email id

d. Validation if no entered any value

e. Re-display for wrongly entered values with message

f. Congratulations and welcome page upon successful entries

OUTCOME

Students will be able to,

1. Create a login page with fields as name, mobile number and email id.
2. Validation of entered data and if wrong data entered proper error messages will be displayed using struts.

SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements:

Operating System: Windows 7/8/10/Ubuntu
Browser: Microsoft Edge/Firefox/Google Chrome
Software: Java, Eclipse

Hardware Requirements:

Processor: Minimum 1 GHz.
Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
Hard Drive: Minimum 32 GB.
Memory (RAM): Minimum 1 GB

THEORY-CONCEPT

Struts 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. When you use Struts, the framework provides you with a controller servlet, ActionServlet, which is defined in the Struts libraries that are included in the IDE, and which is automatically registered in the web.xml deployment descriptor as shown below. The controller servlet uses a struts-config.xml file to map incoming requests to Struts Action objects, and instantiate any ActionForm objects associated with the action to temporarily store form data. The Action object processes requests using it's execute method, while making use of any data stored in the form bean. Once the Action object processes a request, it stores any new data (i.e., in the form bean, or in a separate result bean), and forwards the results to the appropriate view.

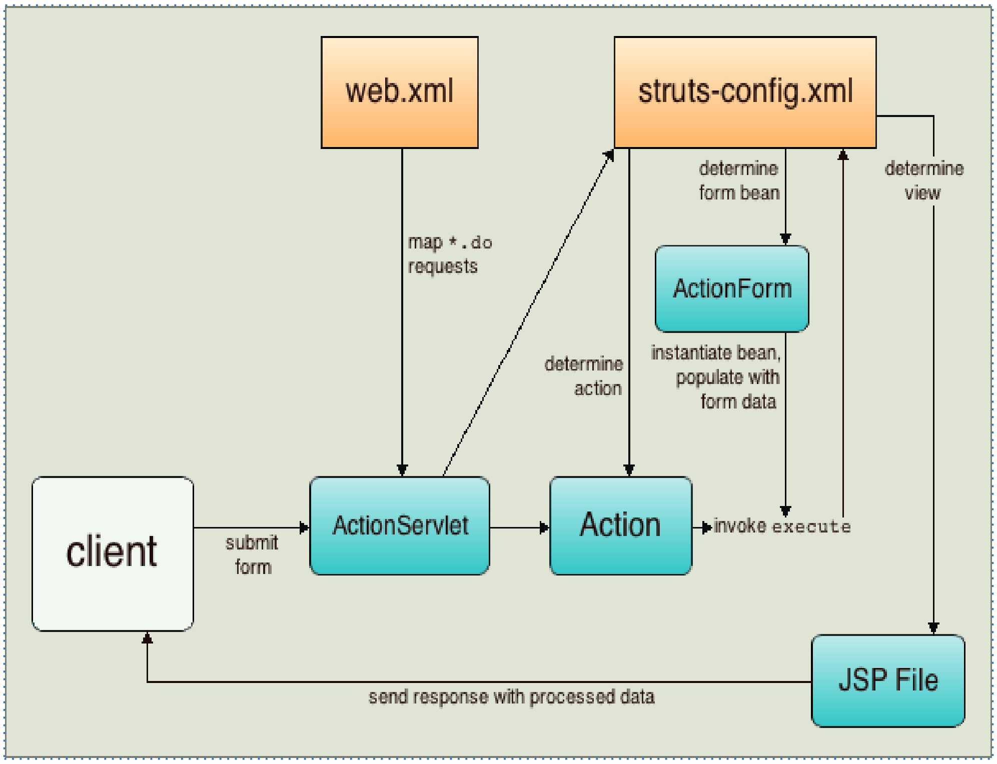


Figure 8.1: Struts workflow

TECHNOLOGY / TOOL

Files Required

Following files are required for this application.

- login.jsp
- success.jsp
- LoginForm.java
- LoginAction.java
- struts-config.xml
- web.xml

Tools used:

In order to create an application we are going to use the following tools.

- JDK 1.5 or above (download)
- Eclipse Indigo or above (download)
- Tomcat 6.x above or any other server which supports java like jboss,weblogic,glassfish (download)
- Struts 1.2 JAR files

DESIGN / EXECUTION STEPS

Step 1: Create Dynamic Web Project

Step 2: Add Jar files to the project

Step 3: Configure web.xml

Step 4: Create FormBean Clas :

Step 5: Create Action Class:

Step 6: Create struts-config.xml file

Step 7: Create jsp files

Step 8: Run the Application

CONCLUSION/ANALYSIS

Hence, we have designed login web page validation using HTML, CSS, JSP, Java and struts.

ORAL QUESTIONS

1. What is Struts?
2. What are the advantages of using Struts?
3. Explain the working flow of Struts?
4. Explain MVC Model of Struts.

Experiment No. - 9

TITLE

Design and develop any web application using AngularJS.

OBJECTIVES

1. Understand the design of single-page applications and how AngularJS facilitates their development
2. Properly separate the model, view, and controller layers of your application and implement them using AngularJS
3. Build Angular forms

PROBLEM STATEMENT

Design an application using AngularJS. e.g., Design registration (first name, last name, username, password) and login page using AngularJS

OUTCOMES

Students can able to,

1. Implement the effective client side implementation.
2. Solve the complex problem of development using MVC framework.

SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements:

1. Operating System: Windows 7/8/10/Ubuntu
2. Browser: Firefox/Google Chrome/ Microsoft Edge etc.
3. Software/Editor : Sublime Editor/Notepad/Notepad++

Hardware Requirements:

1. Processor: Minimum 1 GHz.
2. Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
3. Hard Drive: Minimum 32 GB.
4. Memory (RAM): Minimum 1 GB
5. Sound card-speakers/camera/microphone (Depending upon website selection)

THEORY-CONCEPT

AngularJS is an open-source web application framework. It was initially created in 2009 by Misko Hevery and Adam Abrons. It is presently kept up by Google. Its most recent adaptation is "AngularJS is an auxiliary system for dynamic web applications. It gives you a chance to utilize HTML as your layout dialect and gives you a chance to stretch out HTML's linguistic structure to express your application parts plainly and compactly. Its information official and reliance infusion take out a significant part of the code you as of now need to compose. Also, everything occurs inside the program, making it a perfect band together with any server innovation".

General Features

- AngularJS is a productive system that can make Rich Internet Applications (RIA).
- AngularJS gives designers choices to compose customer side applications utilizing JavaScript in a spotless Model View Controller (MVC) way.
- Applications written in AngularJS are cross-program agreeable. AngularJS consequently handles JavaScript code reasonably for every program.
- AngularJS is open source, totally free, and utilized by a great many engineers the world over. It is authorized under the Apache permit version 2.0.
- By and large, AngularJS is a system to assemble expansive scale, elite, and simple to-keep up web applications.

Core Features:

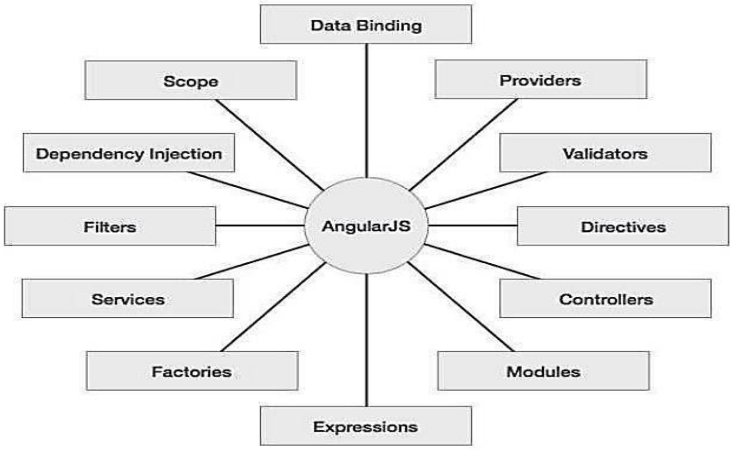


Figure: Architecture of AngularJS

1. **Data-authoritative:** It is the programmed synchronization of information amongst model and view parts.
2. **Scope:** These are objects that allude to the model. They go about as paste amongst controller and view.
3. **Controller:** These are JavaScript capacities bound to a specific degree.
4. **Services:** AngularJS accompanies a few implicit administrations, for example, $http to make aXMLHttpRequests. These are singleton objects which are instantiated just once in application.
5. **Filters:** These select a subset of things from a cluster and restore another exhibit.
6. **Directives:** Directives are markers on DOM components. AngularJS has worked in mandates, for example, ngBind, ngModel, and so on.
7. **Templates:** These are the rendered see with data from the controller and model. These can be a solitary record, (for example, index.html) or different perspectives in a single page utilizing partials.
8. **Routing:** It is idea of exchanging sees**.**
9. **Deep Linking:** Deep connecting permits to encode the condition of use in the URL with the goal that it can be bookmarked. The application would then be able to be re- established from the URL to a similar state.
10. **Dependency Injection:** AngularJS has a worked in reliance infusion subsystem that encourages the designer to make, comprehend, and test the applications effectively.

Advantages of AngularJS

- It gives the ability to make Single Page Application in a spotless and viable way.
- It gives information restricting ability to HTML. Along these lines, it gives client a rich and responsive experience.
- AngularJS code is unit testable.
- AngularJS utilizations reliance infusion and make utilization of partition of concerns.
- AngularJS gives reusable segments.
- With AngularJS, the engineers can accomplish greater usefulness with short code.
- In AngularJS, sees are unadulterated html pages, and controllers written in JavaScript do the business handling.

Model View Controller

Model View Controller or MVC as it is famously called, is a product configuration design for creating web applications. A Model View Controller design is comprised of the accompanying three sections.

☐ Model – It is the most minimal level of the example in charge of looking after information.

□ View – It is in charge of showing all or a part of the information to the client.

□ Controller – It is a product Code that controls the connections between the Model and View.

AngularJS is a MVC based structure.

⊔ **An AngularJS application comprises of following three essential parts –ng-app** – This directive
defines and links an AngularJS application to HTML.

⊔ **ng-model** – This directive binds the values of AngularJS application data to HTML input controls.

⊔ **ng-bind** – This directive binds the AngularJS Application data to HTML tags.

DESIGN/EXECUTION STEPS

Steps for AngularJS

1. *Either download AngularJS*
2. *Or Use CDN Method.*

Example:

We have included the AngularJS JavaScript file in the HTML page so we can use AngularJS –

```
<head>

<scriptsrc="https://ajax.googleapis.com/ajax/libs/angularjs/1.4.8/angular.min.js"></script>

</head>
```

1. *Point to AngularJS app*

Next we tell what part of the HTML contains the AngularJS app. This done by adding the *ng- app* attribute to the root HTML element of the AngularJS app. You can either add it to *html* element or *body* element as shown below –

```
<body ng-app="myapp">

</body>
```

2. *View*

The view is this part –

```
<div ng-controller="HelloController">

<h2>Welcome {{helloTo.title}} to the world of Tutorialspoint!</h2>

</div>
```

ng-controller tells AngularJS what controller to use with this view. *helloTo.title*tells AngularJS

to write the "model" value named helloTo.title to the HTML at this location.

3. *Controller*

The controller part is –

```
<script>
angular.module("myapp",[])

.controller("HelloController",function($scope){

     $scope.helloTo={};

     $scope.helloTo.title="AngularJS";

});
```

This code registers a controller function named *HelloController* in the angular module named *myapp*. The controller function is registered in angular via the angular.module(...).controller(...) function call.

The $scope parameter passed to the controller function is the *model*. The controller function adds a *helloTo* JavaScript object, and in that object it adds a *title* field.

4. *Execution*

Save the above code as *myfirstexample.html* and open it in any browser.

5. *How AngularJS integrates with HTML*

- ng-app directive indicates the start of AngularJS application.
- ng-model directive then creates a model variable named "name" which can be used with the html page and within the div having ng-app directive.
- ng-bind then uses the name model to be displayed in the html span tag

whenever user input something in the text box.

- Closing</div> tag indicates the end of AngularJS application.

AngularJS directives are used to extend HTML. These are special attributes starting with ng- prefix. We're going to discuss following directives −

- **ng-app** − This directive starts an AngularJS Application.
- **ng-init** − This directive initializes application data.
- **ng-model** − This directive binds the values of AngularJS application data to HTML input controls.
- **ng-repeat** − This directive repeats html elements for each item in a collection.

CONCLUSION/ANALYSIS

With the help of this assignment it is helpful to understand features of AngularJS. MVC model structure and its use in advanced web programming is studied.

ORAL QUESTIONS

1. What is AngularJS and what are some of its advantages?
2. What is the Model View Controller (MVC)?
3. What is data binding in AngularJS? How does it relate to the MVC architecture?
4. Explain the concept of scope. How does scope inheritance work in AngularJS?
5. What are directives? Can you explain the functions of the some directives?

Experiment No.: 10

TITLE

Title: EJB.

PROBLEM STATEMENT

Design and implement a business interface with necessary business logic for any web application using EJB. e.g., Design and implement the web application logic for deposit and withdraw amount transactions using EJB.

OBJECTIVES

1. To understand EJB.
2. To design and implement the web application logic for deposit and withdraw amount transactions using EJB

OUTCOME

Students will be able to,

1. To understand EJB.
2. To design and implement the web application logic for deposit and withdraw amount transactions using EJB

SOFTWARE & HARDWARE REQUIREMENTS

Software Requirements:

1. Operating System: Windows 7/8/10/Ubuntu
2. Browser: Microsoft Edge/Firefox/Google Chrome
3. Software: Java, Eclipse

Hardware Requirements:

1. Processor: Minimum 1 GHz.
2. Ethernet connection (LAN) OR a wireless adapter (Wi-Fi)
3. Hard Drive: Minimum 32 GB.
4. Memory (RAM): Minimum 1 GB

THEORY-CONCEPT

What is EJB?

EJB is an acronym for enterprise java bean. It is a specification provided by Sun Microsystems to develop secured, robust and scalable distributed applications.

To get information about distributed applications, visit RMI Tutorial first.

To run EJB application, you need an application server (EJB Container) such as Jboss, Glassfish, Web logic, Web sphere etc. It performs:

a. life cycle management,
b. security,
c. transaction management, and
d. Object pooling.

EJB application is deployed on the server, so it is called server side component also.

EJB is like COM (Component Object Model) provided by Microsoft. But, it is different from Java Bean, RMI and Web Services.

When use Enterprise Java Bean?

1. Application needs Remote Access. In other words, it is distributed.
2. Application needs to be scalable. EJB applications supports load balancing, clustering and fail-over.
3. Application needs encapsulated business logic. EJB application is separated from presentation and persistent layer.

Types

EJB is primarily divided into three categories; following table lists their names with brief descriptions −

S.No	Type & Description
1	**Session Bean** Session bean stores data of a particular user for a single session. It can be stateful or stateless. It is less resource intensive as compared to entity bean. Session bean gets destroyed as soon as user session terminates.

2	**Entity Bean** Entity beans represent persistent data storage. User data can be saved to database via entity beans and later on can be retrieved from the database in the entity bean.
3	**Message Driven Bean** Message driven beans are used in context of JMS (Java Messaging Service). Message Driven Beans can consumes JMS messages from external entities and act accordingly.

Disadvantages of EJB

1. Requires application server
2. Requires only java client. For other language client, you need to go for webservice.
3. Complex to understand and develop ejb applications.

TECHNOLOGY / TOOL

Tools used:

In order to create an application we are going to use the following tools.

- JDK 1.5 or above (download)
- Eclipse Indigo or above (download)
- Tomcat 6.x above or any other server which supports java like jboss,weblogic,glassfish (download)
- EJB (Enterprise Java Bean)

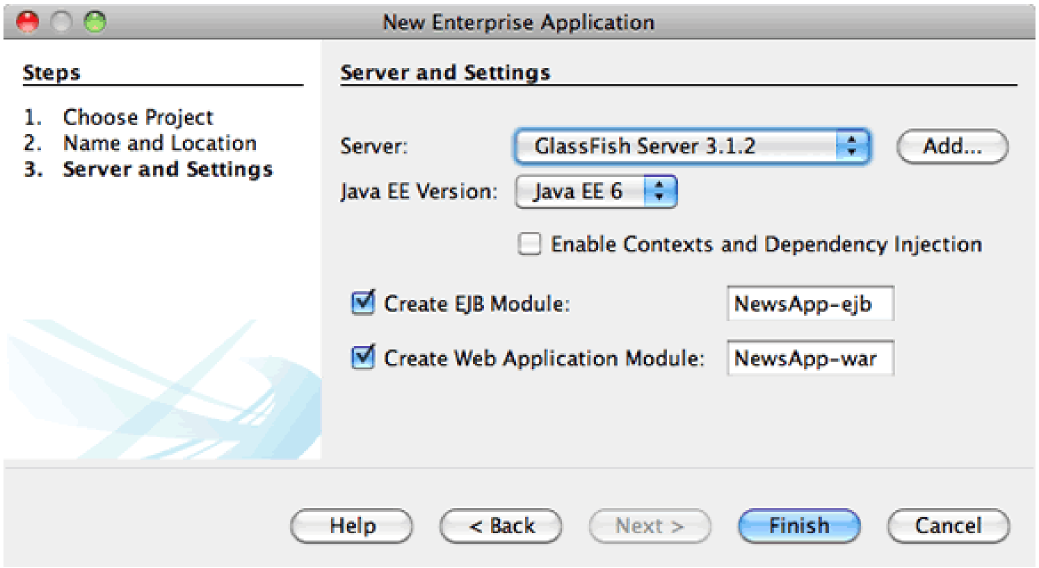
DESIGN / EXECUTION STEPS

Creating the Enterprise Application Project

The goal of this exercise is to create the NewsApp enterprise application project. You will use the New Project wizard to create an enterprise application that contains an EJB module and a web module.

1. Choose File > New Project (Ctrl-Shift-N) from the main menu.
2. Select Enterprise Application from the Java EE category and click Next.

3. Name the project NewsApp and set the project location.
4. Deselect the Use Dedicated Folder option, if selected. (For this tutorial there is little reason to copy project libraries to a dedicated folder because you will not need to share libraries with other users or projects.) Click Next.
1. Set the server to GlassFish Server and set the Java EE Version to Java EE 6 or Java EE 7.
2. Select Create EJB Module and Create Web Application Module. Click Finish.



New Enterprise Application

Steps

1. Choose Project
2. Name and Location
3. **Server and Settings**

Server and Settings

Server: GlassFish Server 3.1.2

Java EE Version: Java EE 6

□ Enable Contexts and Dependency Injection

☑ Create EJB Module: NewsApp-ejb

☑ Create Web Application Module: NewsApp-war

Figure 1. New Project wizard

When you click Finish, the IDE creates three projects: NewsApp, NewsApp-ejb and NewsApp-war. If you expand the NewsApp node in the Projects window, you can see that the enterprise application project does not contain any sources. All the sources will be contained in the two modules that the wizard created and which are listed under the Java EE Modules node.

The enterprise application project only contains configuration and packaging details about the application. When you build and run an enterprise application the IDE creates an EAR archive and deploys the EAR to the server. In some cases, the enterprise application project will contain deployment descriptor files with additional information, but deployment descriptor files are not required when you create a Java EE enterprise application that is deployed to GlassFish Server.

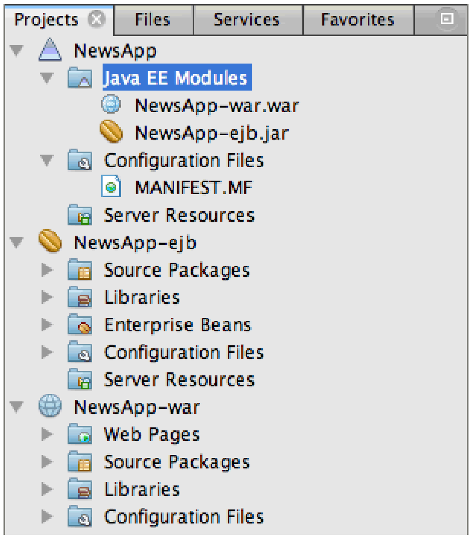


Figure 2. Projects window showing structure of the application

CONCLUSION/ANALYSIS

Hence, designed and implemented the web application logic for deposit and withdraw amount transactions using EJB.

ORAL QUESTIONS

1. What is EJB? &
2. What are Types of EJB?
3. What are the advantages of using EJB?
4. Explain the working flow of EJB?