## application requirements and Specification

1. **Introduction**
   1. Purpose

The main purpose of this project is to write set of simple applications using Servlet and JavaServer Pages (JSP). Servlet technology is used to create web application (resides at server side and generates dynamic web page).

JSP technology is used to create web application just like Servlet technology. It can be thought of as an extension to servlet because it provides more functionality than servlet such as expression language, jstl etc.

* 1. OVERVIEW

Servlet technology is robust and scalable because of java language. Before Servlet, CGI (Common Gateway Interface) scripting language was popular as a server-side programming language. But there was many disadvantages of this technology. We have discussed these disadvantages below.

Servlet can be described in many ways, depending on the context.

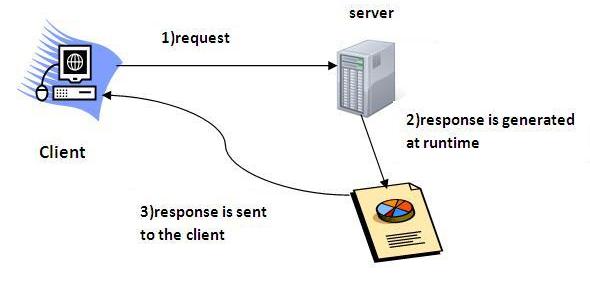
Servlet is a technology i.e. used to create web application.

Servlet is an API that provides many interfaces and classes including documentations.

Servlet is an interface that must be implemented for creating any servlet.

Servlet is a class that extend the capabilities of the servers and respond to the incoming request. It can respond to any type of requests.

Servlet is a web component that is deployed on the server to create dynamic web page.



A JSP page consists of HTML tags and JSP tags. The jsp pages are easier to maintain than servlet because we can separate designing and development. It provides some additional features such as Expression Language, Custom Tag etc.

The JSP pages follows these phases:

Translation of JSP Page

Compilation of JSP Page

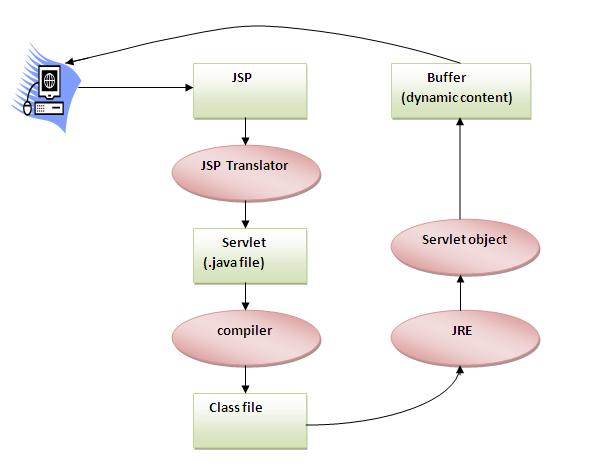
Class loading (class file is loaded by the class loader)

Instantiation (Object of the Generated Servlet is created).

Initialization (jspInit() method is invoked by the container).

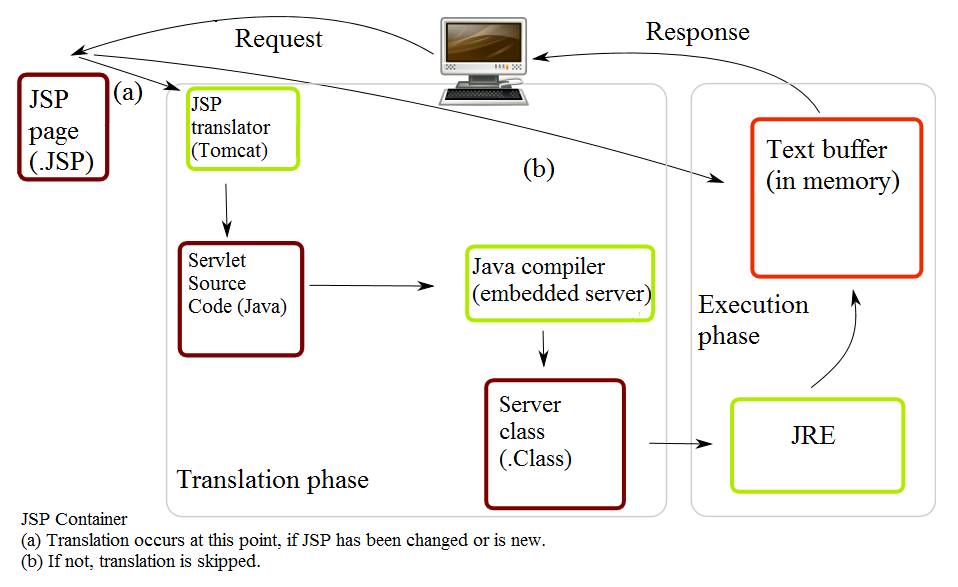
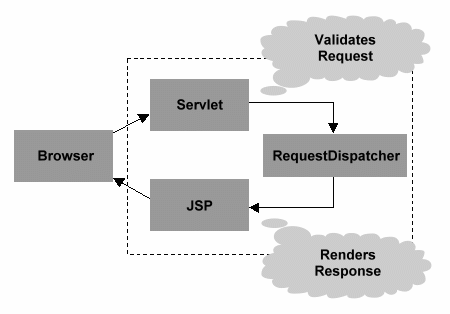
Request processing (\_jspService() method is invoked by the container).

Destroy (jspDestroy() method is invoked by the container).



1. **OVERALL DESCRIPTION**

The following figures showing the Servlet and JSP overall view.



1. **PROGRAM**

// **CookieServlet.java**

Create a servlet class which is basically a controller and its main job is to take the request from user and invoke the appropriate business logic, update the state and call the appropriate JSP file. Only one instance will be created for each servlet except in one special case known as ‘SingleThreadModel’ which we probably talk later. Add servlet-api.jar file to the lib directory under WEB-INF folder, you can get the Java file and other source code from Source link which is located below.

import java.io.IOException;

import java.io.PrintWriter;

import java.util.HashMap;

import java.util.Map;

import javax.servlet.ServletException;

import javax.servlet.annotation.WebInitParam;

import javax.servlet.annotation.WebServlet;

import javax.servlet.http.Cookie;

import javax.servlet.http.HttpServlet;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

/\*\*

\* Servlet implementation class CookieServlet

\*/

/\*

\* @WebServlet(urlPatterns = { "/cookies" }, initParams = {

\*

\* @WebInitParam(name = "cookies", value =

\* "com.oracle.coen235.servlets.CookieServlet", description =

\* "Using cookies to maintain state information") })

\*/

public class CookieServlet extends HttpServlet {

private static final long serialVersionUID = 1L;

/\*\*

\* Default constructor.

\*/

public CookieServlet() {

// TODO Auto-generated constructor stub

}

private final Map books = new HashMap();

public void init() {

books.put("C", "$120");

books.put("C++", "$200");

books.put("Python", "$224");

books.put("Java", "$650");

books.put("Ruby", "$434");

books.put("C#", "$600");

}

/\*\*

\* @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse

\* response)

\*/

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

Cookie cookies[] = request.getCookies();

response.setContentType("text/html");

PrintWriter out = response.getWriter();

out.println("<html>");

out.println("<head>");

out.println("<title>Recommendations</title>");

out.println("</head>");

out.println("<body>");

if (cookies != null && cookies.length != 0) {

out.println("<h1>Recommendations</h1>");

out.println("<p>");

// int total=0;

for (int i = 0; i < cookies.length; i++) {

// out.println( cookies[i].getName() + " How to Program. " +

// "ISBN#: " + cookies[i].getValue() + "<br />" );

out.println(" How to Program in " + cookies[i].getName() + "</br>" + " Price: "

+ cookies[i].getValue() + "<br />");

// total=total+Integer.parseInt(cookies[i].getValue());

// int total=0;

//

// for ( int j = 0 ; j < cookies.length ; j++ ){

//

// total=total+Integer.parseInt(cookies[j].getValue());

//

// }

}

// out.print("Total cost "+ total);

out.println("</p>");

}

else {

out.println("<h1>No Recommendations</h1>");

out.println("<p>You did not select a language.</p>");

}

out.println("</body>");

out.println("</html>");

out.close();

}

/\*\*

\* @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse

\* response)

\*/

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

String language = request.getParameter("language");

String isbn = books.get(language).toString();

Cookie cookie = new Cookie(language, isbn);

response.addCookie(cookie);

response.setContentType("text/html");

PrintWriter out = response.getWriter();

out.println("<html>");

out.println("<head>");

out.println("<title>Welcome to Cookies</title>");

out.println("</head>");

out.println("<body>");

out.println("<p>Welcome to Books order page!! </br></br> You selected " + language + "</p>");

out.println(

"<p><a href = " + "\"CookieSelectLanguage.html\">" + "Click here to choose another book</a></p>");

out.println("<p><a href = " + "\"cookies\">" + "Click here to choose book recommendations</a></p>");

out.println("</body>");

out.println("</html>");

out.close();

}

}

**// web.xml**

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

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns="http://java.sun.com/xml/ns/javaee"

xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"

id="WebApp\_ID" version="3.0">

<display-name>Project5</display-name>

<welcome-file-list>

<welcome-file>index.html</welcome-file>

<welcome-file>index.htm</welcome-file>

<welcome-file>index.jsp</welcome-file>

<welcome-file>default.html</welcome-file>

<welcome-file>default.htm</welcome-file>

<welcome-file>default.jsp</welcome-file>

</welcome-file-list>

<servlet>

<servlet-name>abc</servlet-name>

<servlet-class>com.oracle.coen235.servlets.CookieServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>abc</servlet-name>

<url-pattern>/cookies</url-pattern>

</servlet-mapping>

</web-app>

**// CookieSelectLanguage.html**

At the last but not the least, let us create JSP/html file to give advice back to users. This is a presentation layer which is responsible, only to display the data.

<!-- CookieSelectLanguage.html -->

<html xmlns="http://www.w3c.org/1999/xhtml">

<head>

<title>Using Cookies</title>

</head>

<body>

<form action="cookies" method="post">

<input type="radio" name="language" value="C" />C <br /> <input

type="radio" name="language" value="C++" />C++ <br /> <input

type="radio" name="language" value="Python" />Python <br /> <input

type="radio" name="language" value="Java" />Java <br /> <input

type="radio" name="language" value="Ruby" />Ruby <br /> <input

type="radio" name="language" value="C#" />C# <br /> <input

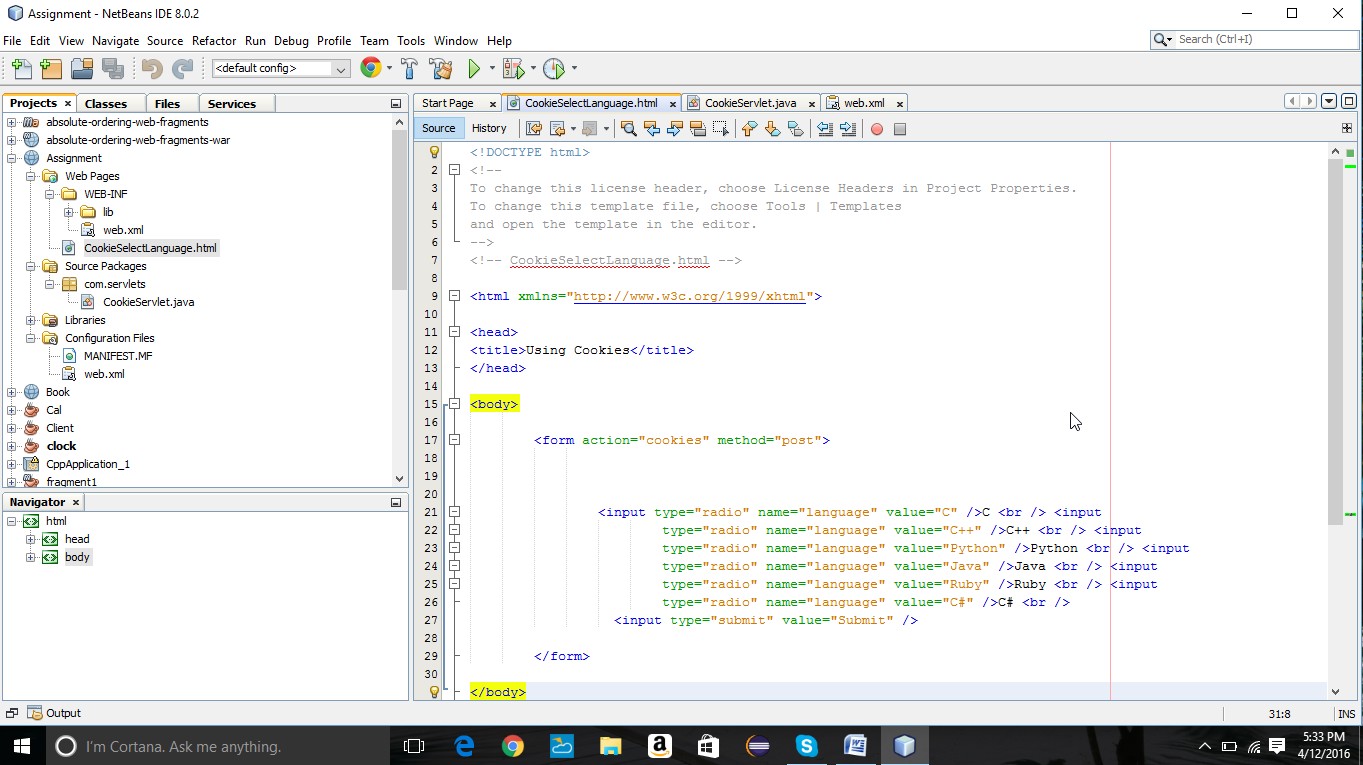
type="submit" value="Submit" />

</form>

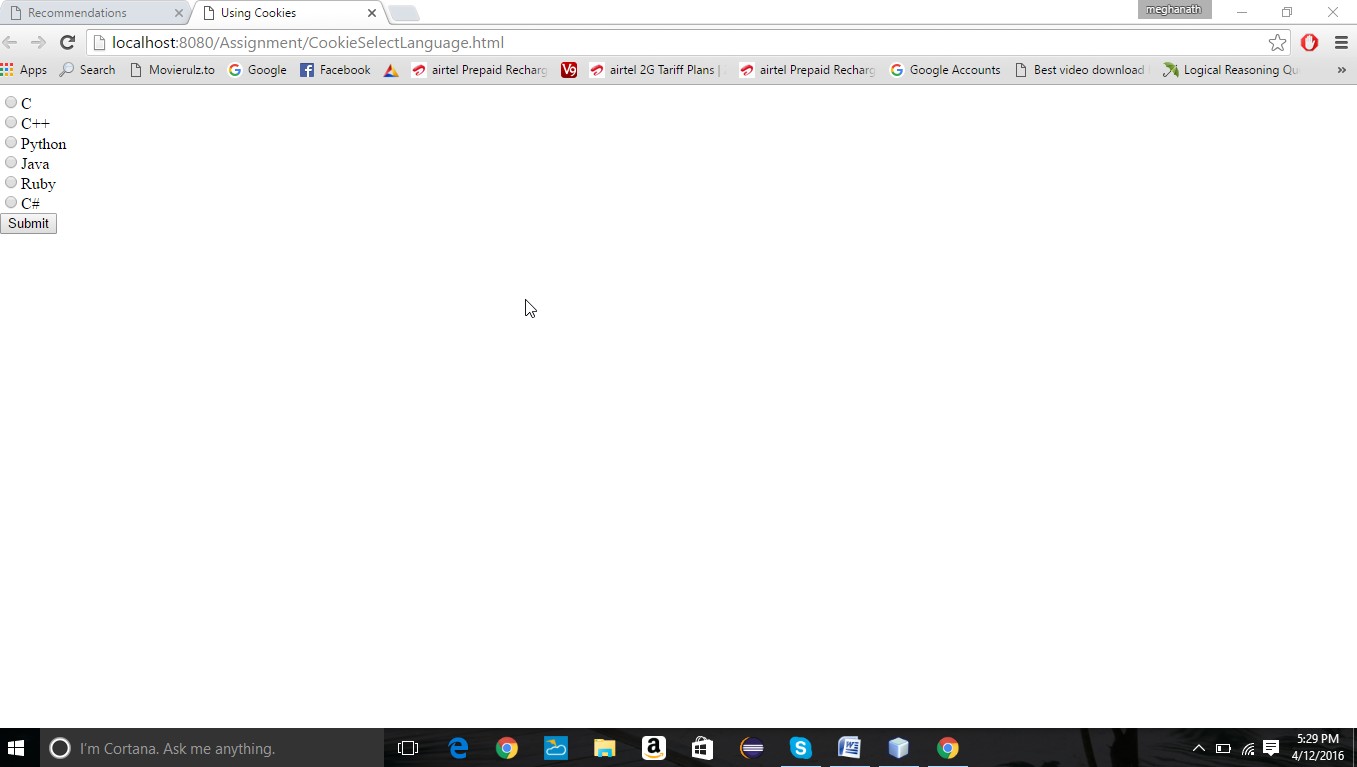
</body>

</html>

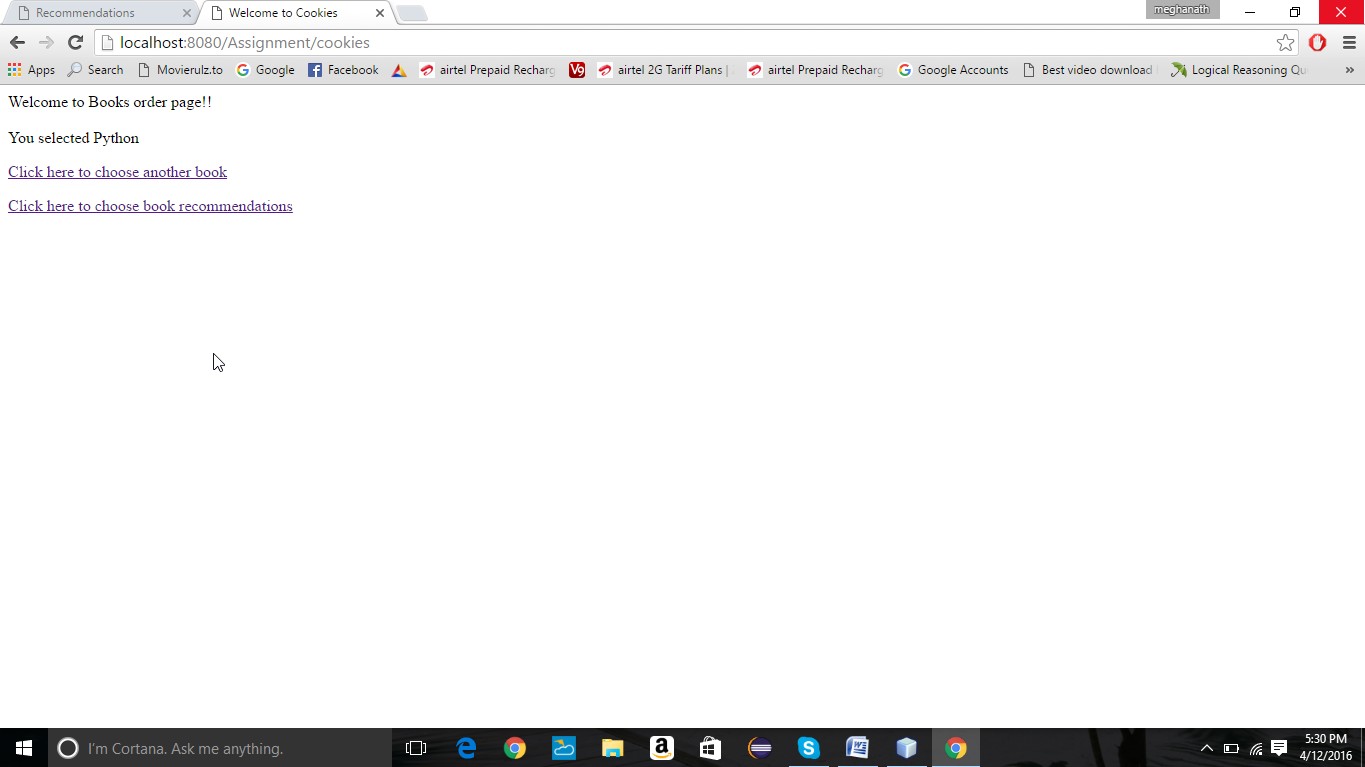
1. **OUTPUT /SCREEN SHOTS**
2. Running the JSP page

****

2.Output on the Client/Web browser to select the books

****

3. Here the options to user to select more books or can view the recommended books.



4. Here the recommended books are displayed.