**Servlet Deployment and GitHub Integration**

Marie Therese Hanna

GITHUB: <https://github.com/marietheresehanna/AD>

**1. Introduction**

This document explains the process of developing and deploying a Java Servlet using Apache Tomcat and IntelliJ IDEA. Additionally, it details the steps for uploading the project to GitHub for version control.

**2. Installation Steps**

**A. Installing Apache Tomcat**

1. Downloaded Apache Tomcat 10.1.34 from the official Apache website.
2. Extracted the Tomcat folder and placed it in C:\tomcat10.
3. Configured IntelliJ IDEA to recognize Tomcat as the application server.

**B. Setting Up IntelliJ IDEA**

1. Created a new **Maven Web Application** project.
2. Chose **Java 22** as the JDK.
3. Selected **Tomcat 10.1.34** as the application server.
4. Set **Jakarta EE 11** as the framework and added the **Servlet** dependency.
5. Chose **Web Application (Servlet, web.xml, index.jsp)** as the project template.
6. Created a package under src/main/java and added a servlet class.
7. Configured web.xml inside src/main/webapp/WEB-INF/.
8. Created index.jsp in src/main/webapp/.

**3. Implementation Details**

**A. Creating the Servlet (HelloServlet.java)**

1. Created a package named com.example.servletproject.
2. Created a Java class named HelloServlet inside src/main/java/com/example/servletproject.
3. Implemented the doGet method to display an HTML message.
4. Used the @WebServlet("/hello") annotation to map the servlet.

**Code for HelloServlet.java**

package com.example.servletproject;

import java.io.IOException;

import java.io.PrintWriter;

import jakarta.servlet.ServletException;

import jakarta.servlet.annotation.WebServlet;

import jakarta.servlet.http.HttpServlet;

import jakarta.servlet.http.HttpServletRequest;

import jakarta.servlet.http.HttpServletResponse;

@WebServlet("/hello")

public class HelloServlet extends HttpServlet {

protected void doGet(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

response.setContentType("text/html");

PrintWriter out = response.getWriter();

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

out.println("<h1>Bonjour, Rima Kilany Chamoun!</h1>");

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

}

}

**B. Configuring web.xml**

1. Opened src/main/webapp/WEB-INF/web.xml.
2. Defined servlet mapping manually to associate the servlet with the URL pattern /hello.

**Code for web.xml**

<web-app xmlns="http://jakarta.ee/xml/ns/jakartaee"

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

xsi:schemaLocation="http://jakarta.ee/xml/ns/jakartaee

http://jakarta.ee/xml/ns/jakartaee/web-app\_5\_0.xsd"

version="5.0">

<servlet>

<servlet-name>HelloServlet</servlet-name>

<servlet-class>com.example.servletproject.HelloServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>HelloServlet</servlet-name>

<url-pattern>/hello</url-pattern>

</servlet-mapping>

</web-app>

**C. Creating index.jsp**

1. Opened src/main/webapp/index.jsp.
2. Created an HTML page with a link to the servlet.

**Code for index.jsp**

<%@ page contentType="text/html;charset=UTF-8" language="java" %>

<html>

<head>

<title>JSP - Hello World</title>

</head>

<body>

<h1>Hello World!</h1>

<a href="hello">Hello Servlet</a>

</body>

</html>

**4. Deployment Process**

**A. Running the Servlet in Tomcat**

1. Clicked **Run > Edit Configurations** in IntelliJ IDEA.
2. Selected **Tomcat Server > Local** and configured deployment.
3. Set the **Artifact** to ServletProject:war exploded.
4. Clicked **Apply** and **Run** to start the Tomcat server.
5. Opened a browser and navigated to http://localhost:8080/ServletProject\_war\_exploded/.
6. Clicked on the **Hello Servlet** link, which directed to http://localhost:8080/ServletProject\_war\_exploded/hello.
7. Verified that the page displayed **"Bonjour, Rima Kilany Chamoun!"**.

**5. Conclusion**

In this project, I successfully set up a **Java Servlet** using **Apache Tomcat** and **IntelliJ IDEA**. The servlet was deployed and tested on a local Tomcat server. I also integrated Git and GitHub to track project versions.

Through this process, I learned:

1. How to configure and deploy a servlet in Tomcat.
2. How to set up IntelliJ IDEA for web development.
3. How to use Git and GitHub for version control.

This project helped reinforce my understanding of **web application development, server-side programming, and Git workflows**.

