**\*\*\*\*\* STEP BY STEP TO BUILD A PROJECT**

Step 1: Create maven project

Step 2: Change project as Dynamic Web and add Runtime environment

Step 3: Add libraries in pom.xml

Step 4: Create web.xml

Step 5: Create spring-webflow-config.xml

Step 6: Config spring-mvc-servlet.xml for Spring

Step 7: Create HelloController.java to handle request

Step 8: Create view

Step 9: Copy libraries to src/main/java/webapp/WEB-INF/lib

jsp-api-2.0.jar

servlet-api-2.4.jar

spring-aop-5.0.2.RELEASE.jar

spring-beans-5.0.2.RELEASE.jar

spring-context-5.0.2.RELEASE.jar

spring-core-5.0.2.RELEASE.jar

spring-expression-5.0.2.RELEASE.jar

spring-jcl-5.0.2.RELEASE.jar

spring-web-5.0.2.RELEASE.jar

spring-webmvc-5.0.2.RELEASE.jar

**\*\*\*\*\* CONFIGURATION MAPPING BY COLORS**

**spring-webflow-config.xml**

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

<beans>

<!-- ================================================================== -->

<!-- Spring Web Flow stuff -->

<!-- ================================================================== -->

<!-- This creates an XmlFlowRegistryFactory bean -->

<bean id=*"viewFactoryCreator"*

class=*"org.springframework.webflow.mvc.builder.MvcViewFactoryCreator"*>

<property name=*"viewResolvers"* ref=*"viewResolverFM"* />

</bean>

<webflow:flow-builder-services

id=*"flowBuilderServices"* view-factory-creator=*"viewFactoryCreator"* />

<webflow:flow-registry id=*"*activationFlowRegistry*"*

flow-builder-services=*"flowBuilderServices"*>

<webflow:flow-location

id="activationFlow"

path=*"/WEB-INF/flows/activation-flow.xml"* />

</webflow:flow-registry>

<webflow:flow-executor id=*"activationFlowExecutor"*

flow-registry=*"*activationFlowRegistry*"*>

</webflow:flow-executor>

<!-- Enables FlowHandler URL mapping -->

<bean

class=*"org.springframework.webflow.mvc.servlet.FlowHandlerAdapter"*>

<property name=*"flowExecutor"* ref=*"activationFlowExecutor"* />

</bean>

<!-- Maps request paths to flows in the flowRegistry; e.g. a path of /hotels/booking

looks for a flow with id "hotels/booking" -->

<bean

class=*"org.springframework.webflow.mvc.servlet.FlowHandlerMapping"*>

<property name=*"flowRegistry"* ref=*"*activationFlowRegistry*"* />

<property name=*"order"* value=*"0"* />

</bean>

</beans>

**spring-mvc-servlet.xml**

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

<beans xmlns=*"http://www.springframework.org/schema/beans"*

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

xmlns:context=*"http://www.springframework.org/schema/context"*

xsi:schemaLocation=*"http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd*

*http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context-4.3.xsd"*>

<context:component-scan

base-package=*"stackjava.com.springmvchello"* />

<!-- START: jsp config -->

<!-- <bean id="jspConfig" class="org.springframework.web.servlet.view.InternalResourceViewResolver">

<property name="prefix"> <value>/WEB-INF/views/jsp/</value> </property>

<property name="suffix"> <value>.jsp</value> </property> </bean> -->

<!-- View resolvers can also be configured with ResourceBundles or XML files.

If you need different view resolving based on Locale, you have to use the

resource bundle resolver. -->

<!-- <bean id="viewResolverJSP" class="org.springframework.web.servlet.view.InternalResourceViewResolver">

<property name="prefix" value="" /> <property name="suffix" value=".jsp"

/> </bean> -->

<!-- END: jsp config -->

<!-- START: freemarker config -->

<bean id=*"freemarkerConfig"*

class=*"org.springframework.web.servlet.view.freemarker.FreeMarkerConfigurer"*>

<property name=*"templateLoaderPath"*

value=*"/WEB-INF/views/freemarker/"* />

</bean>

<!-- View resolvers can also be configured with ResourceBundles or XML files.

If you need different view resolving based on Locale, you have to use the

resource bundle resolver. -->

<bean id=*"viewResolverFM"*

class=*"org.springframework.web.servlet.view.freemarker.FreeMarkerViewResolver"*>

<property name=*"prefix"* value=*""* />

<property name=*"suffix"* value=*".fm"* />

</bean>

<!-- END: freemarker config -->

<!-- START import spring-webflow config -->

<import resource=*"spring-webflow-config.xml"*/>

<!-- END import spring-webflow config -->

</beans>

**index.fm**

<!DOCTYPE html>

<html>

<head>

<title>Demo Spring MVC FreeMarker, Spring Webflow</title>

</head>

<body>

<h2>Click to activate account</h2>

<form method="post" action="${flowExecutionUrl}">

<input type="submit" name="\_eventId\_activate" value="activate" />

<input type="submit" name="\_eventId\_cancel" value="cancel" />

</form>

</body>

</html>

**activation-flow.xml**

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

<flow xmlns=*"http://www.springframework.org/schema/webflow"*

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

xsi:schemaLocation=*"http://www.springframework.org/schema/webflow http://www.springframework.org/schema/webflow/spring-webflow-2.0.xsd"*>

<view-state id=*"****index****"*>

<transition on=*"activate"* to=*"success"* />

<transition on=*"cancel"* to=*"failure"* />

</view-state>

<view-state id=*"success"* />

<view-state id=*"failure"* />

</flow>

**In folder \WEB-INF\views\freemarker**

**index.fm**

**failure.fm**

**success.fm**

**TEST RESULT:** [**http://localhost:8080/SpringMVCHello/activationFlow**](http://localhost:8080/SpringMVCHello/activationFlow)

****

**Spring Webflow Diagram**

productList.fm

Click activate button

success.fm

index.fm

Click cancel

failure.fm

**TROUBLESHOOTS:**

**TBS 1:**

SEVERE: Servlet [spring-mvc] in web application [/SpringMVCHello] threw load() exception java.lang.ClassNotFoundException: org.springframework.web.servlet.DispatcherServlet at org.apache.catalina.loader.WebappClassLoaderBase.loadClass(WebappClassLoaderBase.java:1407) at org.apache.catalina.loader.WebappClassLoaderBase.loadClass(WebappClassLoaderBase.java:1215) at org.apache.catalina.core.DefaultInstanceManager.loadClass(DefaultInstanceManager.java:538) at …

**TBS1 - Solution:**

Copy all jar files to WEB-INF/lib

jsp-api-2.0.jar

servlet-api-2.4.jar

spring-aop-5.0.2.RELEASE.jar

spring-beans-5.0.2.RELEASE.jar

spring-context-5.0.2.RELEASE.jar

spring-core-5.0.2.RELEASE.jar

spring-expression-5.0.2.RELEASE.jar

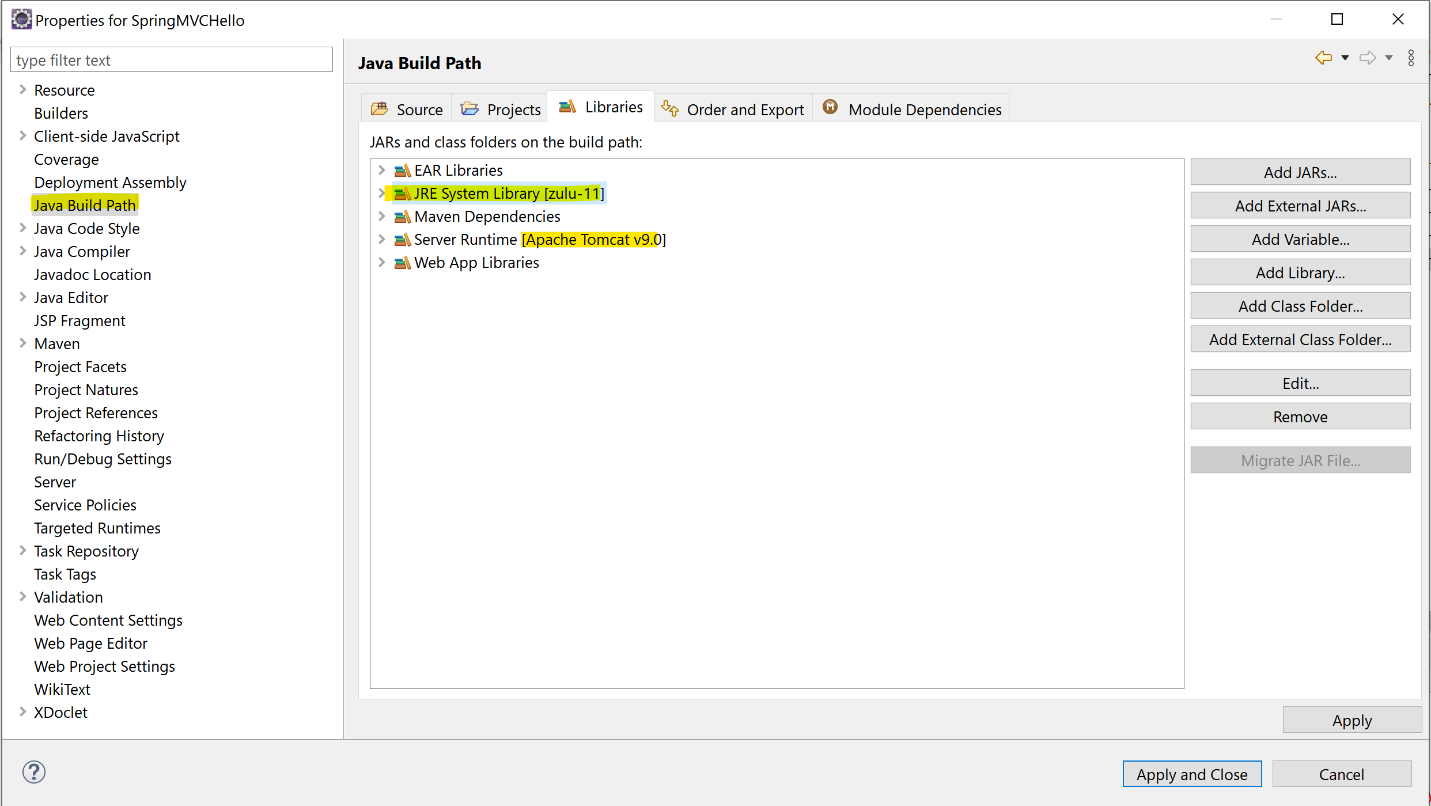
spring-jcl-5.0.2.RELEASE.jar

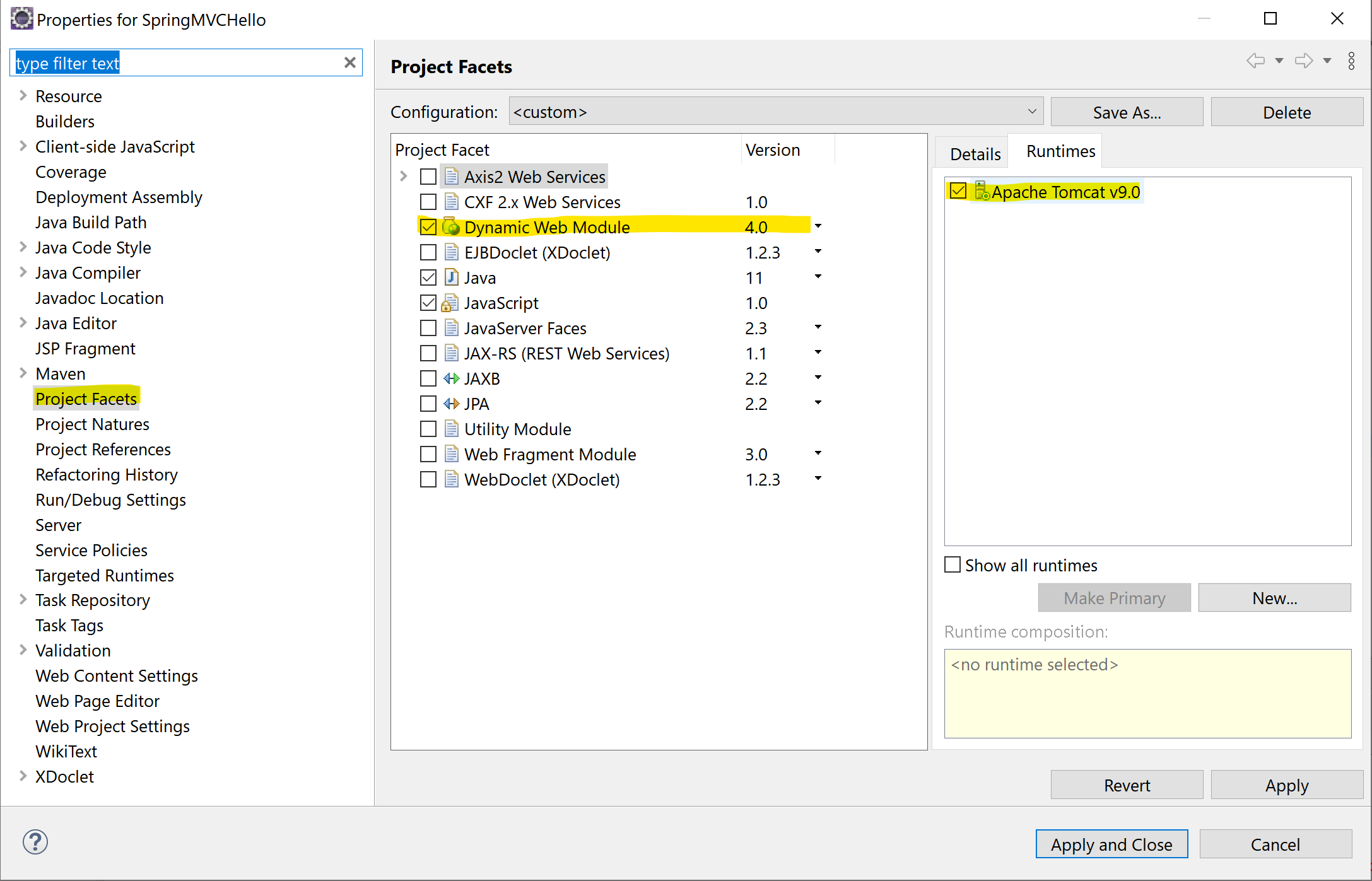
spring-web-5.0.2.RELEASE.jar

spring-webmvc-5.0.2.RELEASE.jar

**TBS 2: Dynamic Web Module 3.0 requires Java 1.6 or newer**

**TBS 2 - Solution:**

Go to **Properties** and check **Java Build Path** and **Project facets** 



**TBS3:**

java.io.NotSerializableException: stackjava.com.springmvchello.model.Product

at java.base/java.io.ObjectOutputStream.writeObject0(ObjectOutputStream.java:1185)

at java.base/java.io.ObjectOutputStream.writeObject(ObjectOutputStream.java:349)

at java.base/java.util.ArrayList.writeObject(ArrayList.java:897)

at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

**TBS3 - Solution: Add ` implements** Serializable`

**public** **class** Product **implements** Serializable {

/\*\*

\*

\*/

}

**REFERENCES :**

[Guide to Spring Web Flow | Baeldung](https://www.baeldung.com/spring-web-flow)

[10. System Setup (spring.io)](https://docs.spring.io/spring-webflow/docs/current/reference/html/system-setup.html)

[Code ví dụ Spring MVC FreeMarker Hello World + Maven + Eclipse (stackjava.com)](https://stackjava.com/spring-mvc/code-vi-du-spring-mvc-freemarker-hello-world-maven-eclipse.html)