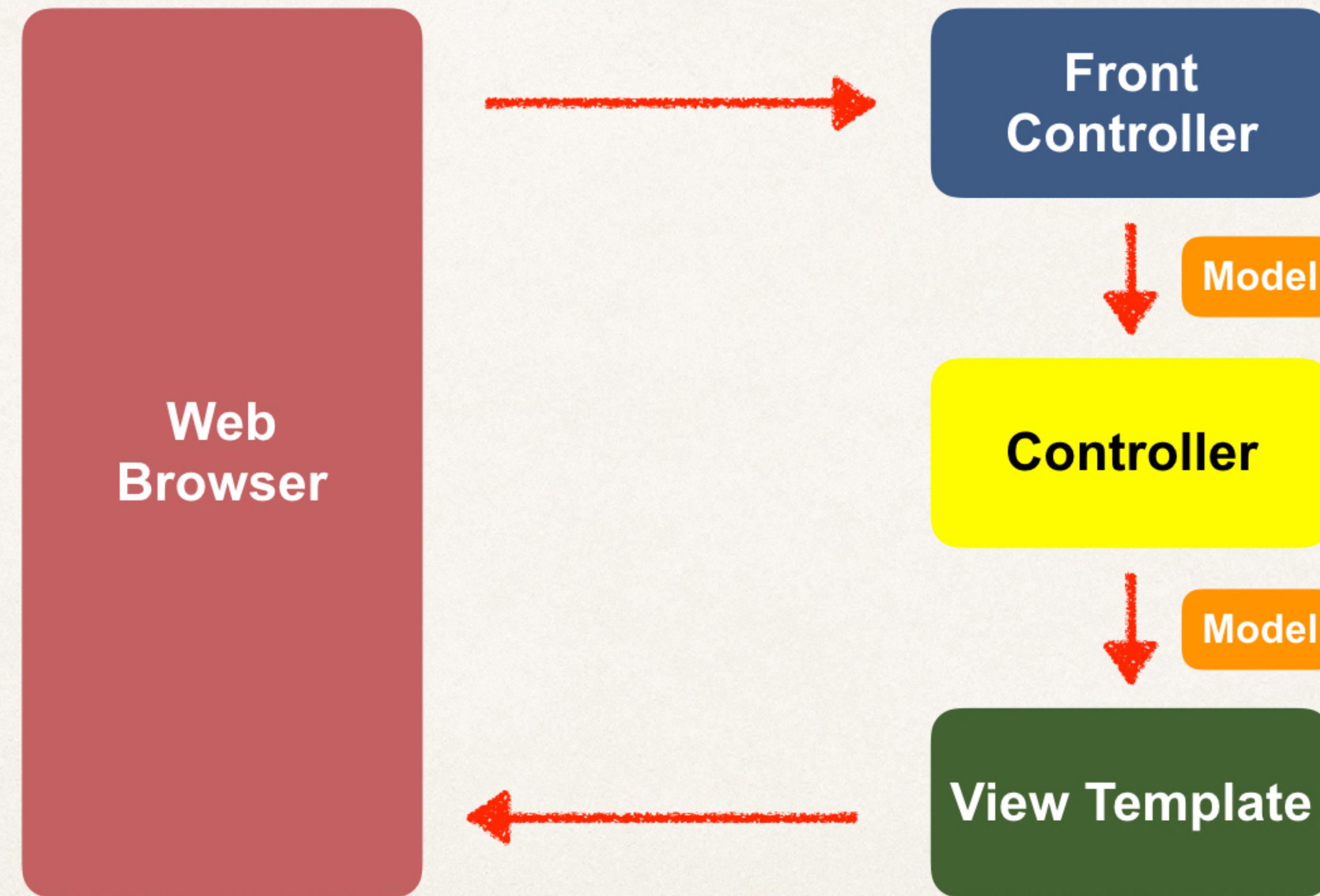


SPRING MVC CONFIGURATION



Model-View-Controller (MVC)



Spring MVC Configuration Process - Part 1

Add configurations to file: **WEB-INF/web.xml**

Step-By-Step

1. Configure Spring MVC Dispatcher Servlet
2. Set up URL mappings to Spring MVC Dispatcher Servlet

Spring MVC Configuration Process - Part 2

Add configurations to file: **WEB-INF/spring-mvc-demo-servlet.xml**

3. Add support for Spring component scanning
4. Add support for conversion, formatting and validation
5. Configure Spring MVC View Resolver

Step-By-Step

Step 1: Configure Spring DispatcherServlet

File: web.xml

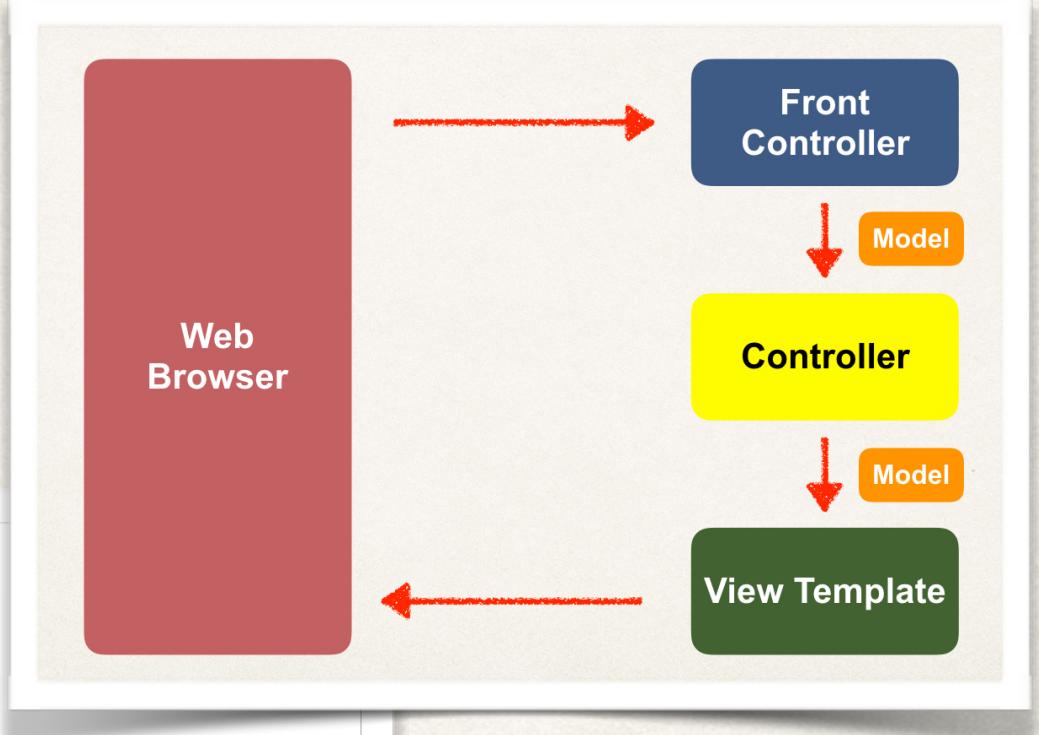
```
<web-app>

    <servlet>
        <servlet-name>dispatcher</servlet-name>
        <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

        <init-param>
            <param-name>contextConfigLocation</param-name>
            <param-value>/WEB-INF/spring-mvc-demo-servlet.xml</param-value>
        </init-param>

        <load-on-startup>1</load-on-startup>
    </servlet>

</web-app>
```



Step 2: Set up URL mappings to Spring MVC Dispatcher Servlet

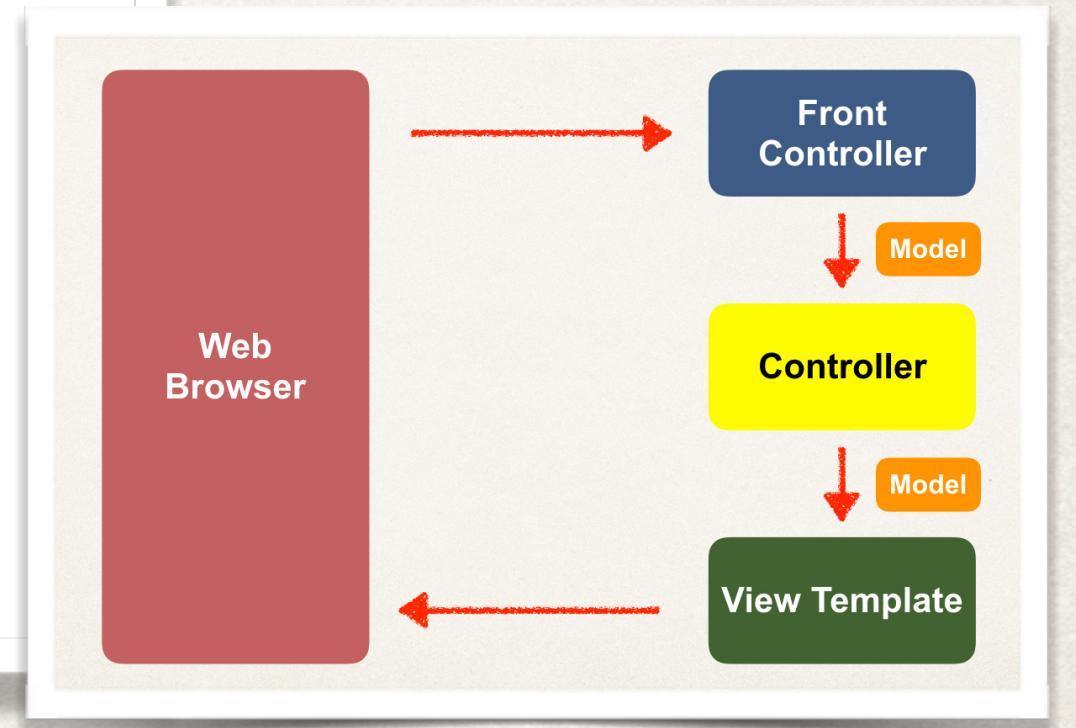
File: web.xml

```
<web-app>

    <servlet>
        <servlet-name>dispatcher</servlet-name>
        <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
        ...
    </servlet>

    <servlet-mapping>
        <servlet-name>dispatcher</servlet-name>
        <url-pattern>/</url-pattern>
    </servlet-mapping>

</web-app>
```



Step 3: Add support for Spring component scanning

File: spring-mvc-demo-servlet.xml

```
<beans>

    <!-- Step 3: Add support for component scanning -->
    <context:component-scan base-package="com.luv2code.springdemo" />

</beans>
```

Step 4: Add support for conversion, formatting and validation

File: spring-mvc-demo-servlet.xml

```
<beans>

    <!-- Step 3: Add support for component scanning -->
    <context:component-scan base-package="com.luv2code.springdemo" />

    <!-- Step 4: Add support for conversion, formatting and validation support -->
    <mvc:annotation-driven/>

</beans>
```

Step 5: Configure Spring MVC View Resolver

File: spring-mvc-demo-servlet.xml

```
<beans>

    <!-- Step 3: Add support for component scanning -->
    <context:component-scan base-package="com.luv2code.springdemo" />

    <!-- Step 4: Add support for conversion, formatting and validation support -->
    <mvc:annotation-driven/>

    <!-- Step 5: Define Spring MVC view resolver -->
    <bean
        class="org.springframework.web.servlet.view.InternalResourceViewResolver">
        <property name="prefix" value="/WEB-INF/view/" />
        <property name="suffix" value=".jsp" />
    </bean>

</beans>
```

View Resolver Configs - Explained

- When your app provides a “view” name, Spring MVC will
 - prepend the prefix
 - append the suffix

```
<bean  
    class="org.springframework.web.servlet.view.InternalResourceViewResolver">  
    <property name="prefix" value="/WEB-INF/view/" />  
    <property name="suffix" value=".jsp" />  
</bean>
```

View Resolver Configs - Explained

```
<bean  
    class="org.springframework.web.servlet.view.InternalResourceViewResolver"  
    <property name="prefix" value="/WEB-INF/view/" />  
    <property name="suffix" value=".jsp" />  
</bean>
```

/WEB-INF/view/ show-student-list .jsp



View name

SPRING MVC CONFIGURATION

