**Deploying Spring Boot Application on External Server**

The following document will explain how we deployed our Spring Boot application on an external tomcat server. Apache Tomcat 8 (i.e. apache-tomcat-8.5.75) was used using the .war file for the deployment, with spring boot 2.7.4 release.

**The following need to be taken into consideration for deploying the Spring Boot Application on External Server : -**

* Mention the external tomcat dependencies in pom.xml to stop the internal tomcat during execution.
* Build the .war file with all necessary dependencies inside build tag to build .war file with custom name as follows : <finalName>RetailDelight</finalName>
* The URL : [Retail Delight](http://192.168.141.86:8080/RetailDelight/) here is the file name RetailDelight.war of .war file.

**The following steps need to be followed to deploying our Spring Boot application on an external tomcat : -**

1. Add the dependency shown below in the pom.xml to exclude embedded tomcat so that we can further deploy our war file on external tomcat server.

Text

Description automatically generated

1. Define the Spring Main class by extending Spring Boot Servlet Initializer class and then override the configure() method.

Graphical user interface, text, application, email

Description automatically generated

1. Mention the packaging as war and build the .war file application using maven build command.

Graphical user interface, text, application

Description automatically generated

1. The War file should be in the target folder.

Graphical user interface

Description automatically generated

1. The downloaded Tomcat 8.5 version after extracting should like this as shown below.

Graphical user interface, application

Description automatically generated

1. The war file should then be placed in the webapps folder of extracted Tomcat 8.5 version.

Graphical user interface, text, application

Description automatically generated

1. Open the bin folder of extracted Tomcat 8.5 version and then run the startup.bat file.

Table

Description automatically generated

1. Then configure the tomcat-user file to add the users to manage the apps after deployment.

Text

Description automatically generated

1. Start the Tomcat 8.5 server using command startup.bat in command prompt which will start tomcat server and also run the .war file present in webapps folder.
2. Then open browser and go to the URL : localhost:8080/ to open Apache Tomcat/8.5.82 dashboard.

Graphical user interface, text, application, website

Description automatically generated

1. Click on the Manager App button and start the spring application on tomcat server.

Graphical user interface, application

Description automatically generated with medium confidence

1. Click on the application URL : [Retail Delight](http://192.168.141.86:8080/RetailDelight/) (i.e. war file name in webapps).

Table

Description automatically generated

**With this our application, Retail Delight has been successfully deployed on the External Tomcat server using Tomcat 8.5 version.**