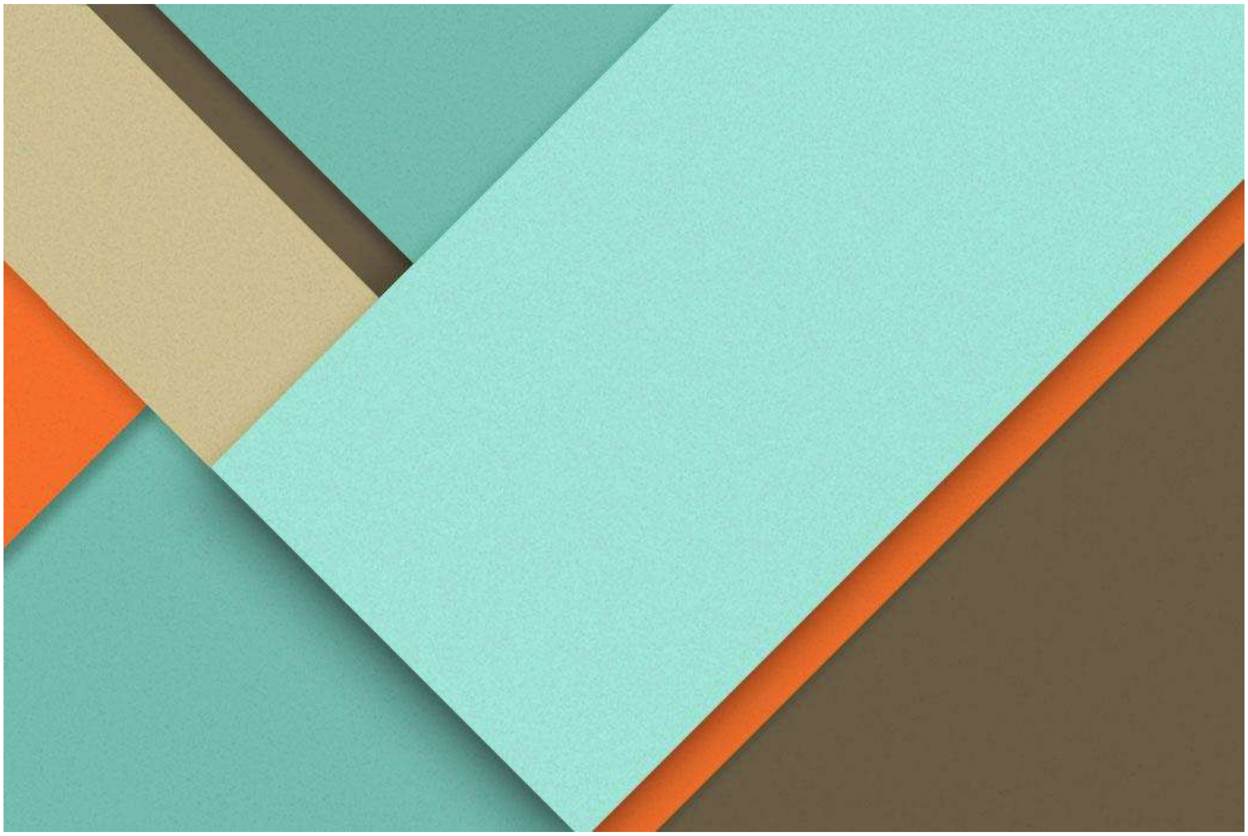


Time Now

A Java web app that is built using Spring Boot and deployed in Amazon EC2 using Jenkins CI/CD pipeline. It is a simple app, that can display the current date and time in US East Zone.

Rajagopal Krishnasamy

rajagopal016@gmail.com

Project Github : https://github.com/rajagopal016/simpleproject.git

Overview

This project is aimed to create a Spring Boot app that can display the current date and time. It is deployed in Tomcat using Jenkins CI/CD pipeline, hosted in AWS EC2 VM.

Goals

1. Create a Spring Boot App, that can display current date and time
2. Create a AWS EC2 VM
3. Create a Jenkins pipeline that can perform CI and CD operations

Specifications

1. A Java web app, that can display date and time of US East region

   Tools Used:

   i. Java 16
   ii. Apache Maven 3.8
   iii. Spring Boot 2.6.1
   iv. GitHub for version controlling the project
   https://github.com/rajagopal016/simpleproject.git
2. AWS EC2 VM was created to host the Jenkins pipeline run and to deploy the app in the Apache Tomcat Webserver. A Windows Server 2019 VM was created and the following SW were installed to realise the goals
   i. VM OS - Windows Server 2019
   ii. Apache Tomcat 9.0
   iii. Apache Maven 3.8.4
   iv. Jenkins 2.325

Milestones

I. Sprint - 1

1. Understand the requirements of the project
2. Create Web App to display the time and date

3. Test the app

II. Sprint - 2

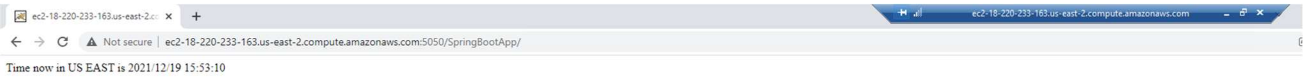
1. Create an AWS EC2 VM - Windows Server 2019
2. Install above mentioned Software
3. Start Jenkins(port 8080) and Tomcat(port 5050)

III. Sprint - 3

4. Create a Jenkins pipeline to perform CI/CD operation
5. Test the pipeline
6. Perform final testing of the deployed app

Web Application Details :

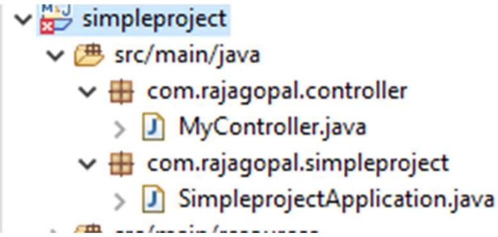
Time Now is a Spring Java App, that can display the system date and time. It can be accessed over web and it displays the time and date instantly.


ec2-18-220-233-163.us-east-2.c
ec2-18-220-233-163.us-east-2.compute.amazonaws.com
Not secure

ec2-18-220-233-163.us-east-2.compute.amazonaws.com:5050/SpringBootApp/

Time now in US EAST is 2021/12/19 15:53:10

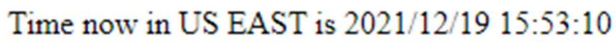
Screenshot of the date and time shown by the app



```
simpleproject
  src/main/java
    com.rajagopal.controller
      MyController.java
    com.rajagopal.simpleproject
      SimpleprojectApplication.java
```

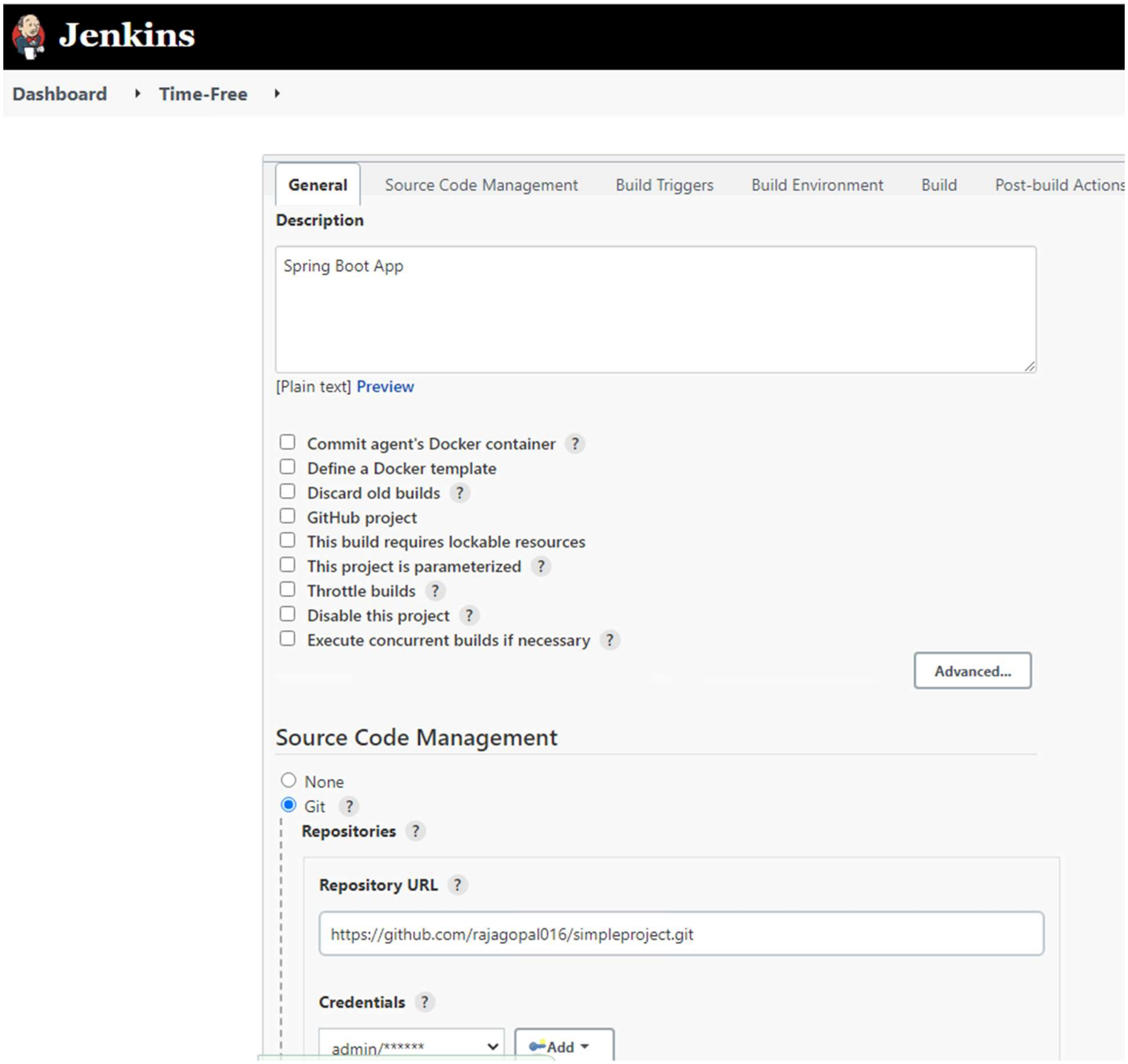
Screenshot of class structure of the app

The homepage http://ec2-18-220-233-163.us-east-2.compute.amazonaws.com:5050/SpringBootApp/ will display the date and time as



Time now in US EAST is 2021/12/19 15:53:10

Jenkins Pipeline



Jenkins
Dashboard ▸ Time-Free ▸
General | Source Code Management | Build Triggers | Build Environment | Build | Post-build Actions

Description

Spring Boot App

[Plain text] Preview

- □ Commit agent's Docker container ?
- □ Define a Docker template
- □ Discard old builds ?
- □ GitHub project
- □ This build requires lockable resources
- □ This project is parameterized ?
- □ Throttle builds ?
- □ Disable this project ?
- □ Execute concurrent builds if necessary ?

Source Code Management

○ None
● Git ?

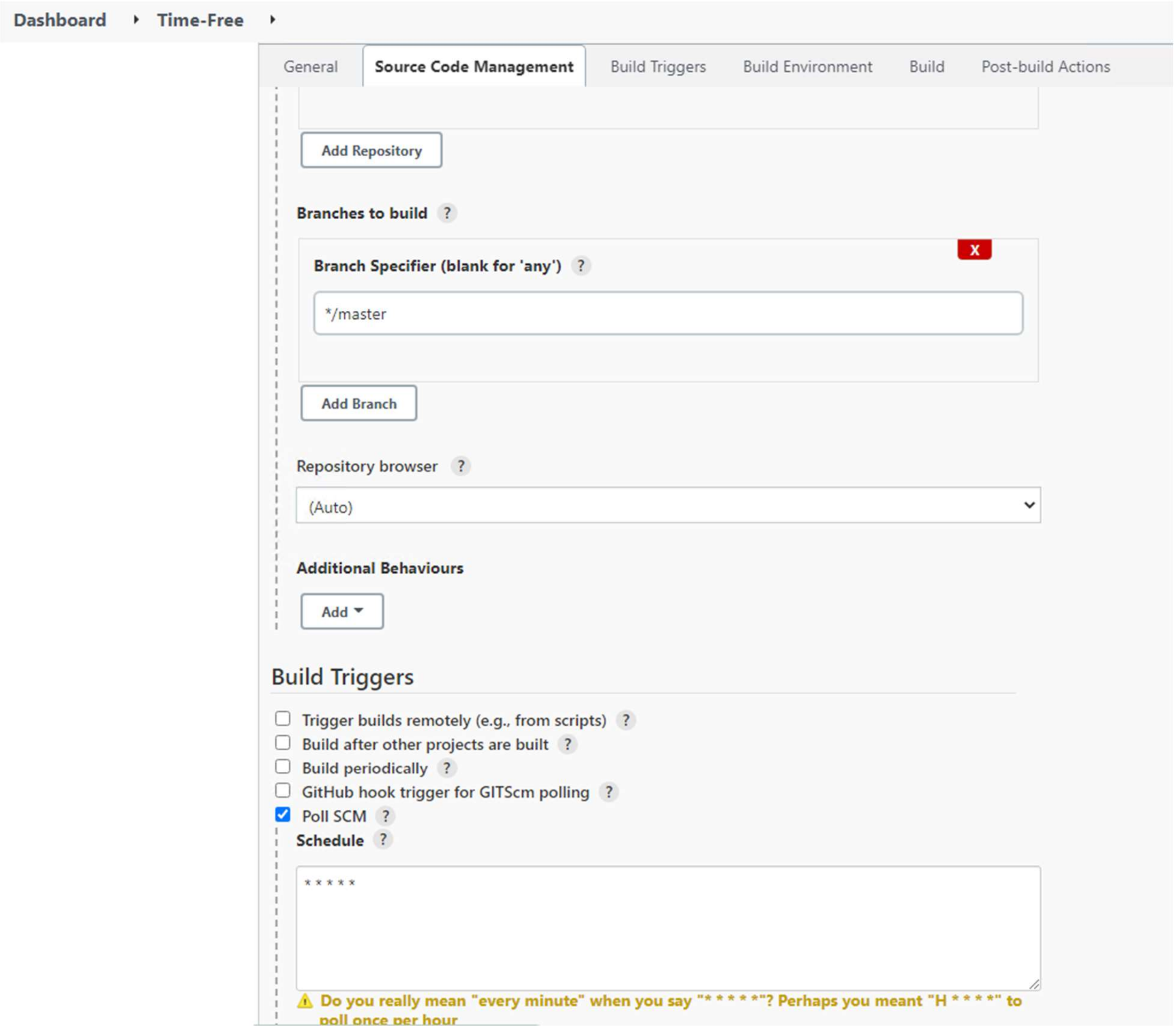
Repositories ?

Repository URL ?

https://github.com/rajagopal016/simpleproject.git

Credentials ?

admin/******



Branches to build ?

Branch Specifier (blank for 'any') ?

*/master

Repository browser ?

(Auto)

Additional Behaviours

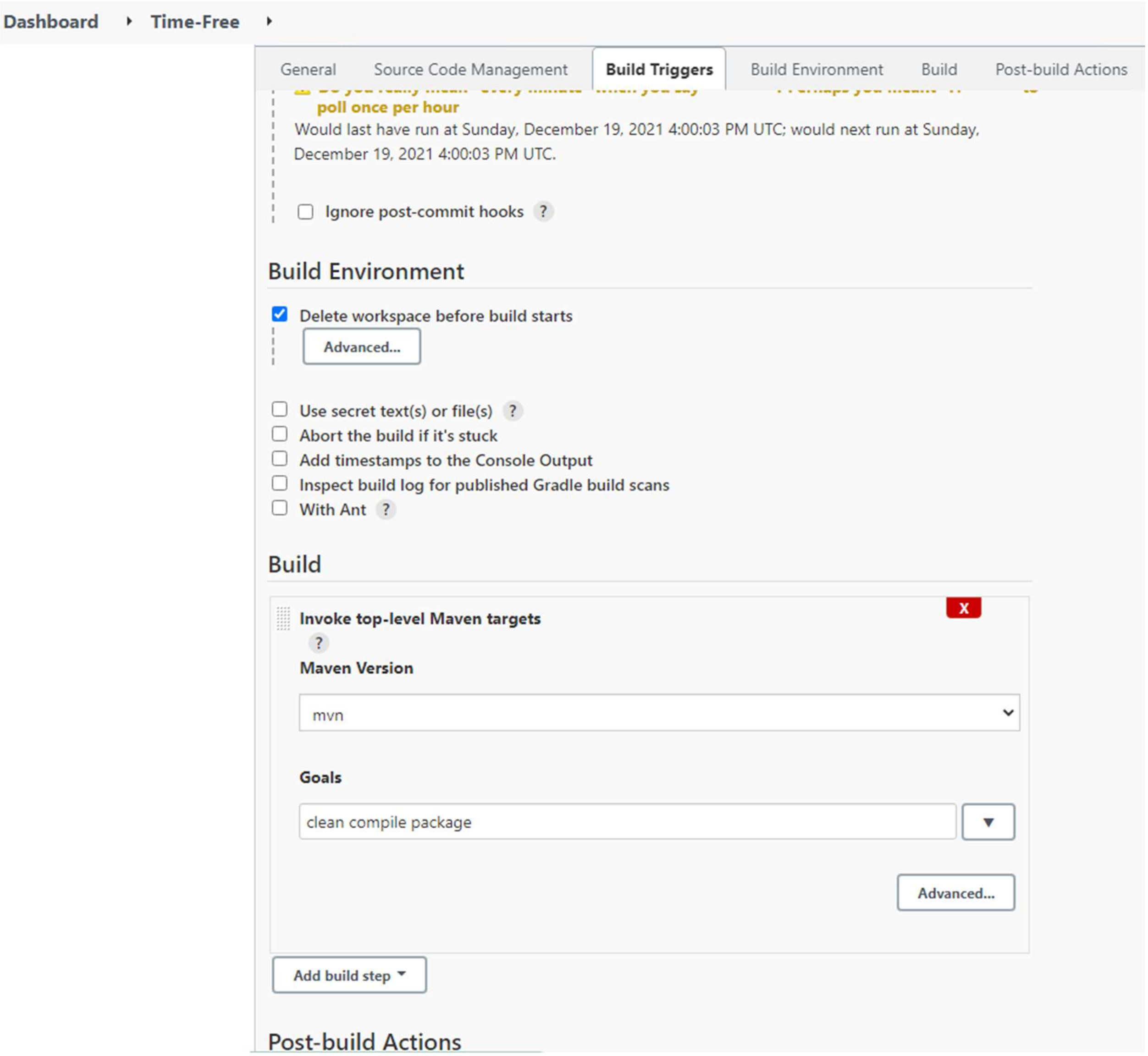
Build Triggers

- ☐ Trigger builds remotely (e.g., from scripts) ?
- ☐ Build after other projects are built ?
- ☐ Build periodically ?
- ☐ GitHub hook trigger for GITScm polling ?
- ☑ Poll SCM ?

Schedule ?

```
* * * * *
```

⚠ Do you really mean "every minute" when you say "* * * * *"? Perhaps you meant "H * * * *" to poll once per hour



Dashboard ▸ Time-Free ▸

⚠ Do you really mean "every minute" when you say ? Perhaps you meant "H to poll once per hour

Would last have run at Sunday, December 19, 2021 4:00:03 PM UTC; would next run at Sunday, December 19, 2021 4:00:03 PM UTC.

□ Ignore post-commit hooks ?

Build Environment

☑ Delete workspace before build starts

□ Use secret text(s) or file(s) ?
□ Abort the build if it's stuck
□ Add timestamps to the Console Output
□ Inspect build log for published Gradle build scans
□ With Ant ?

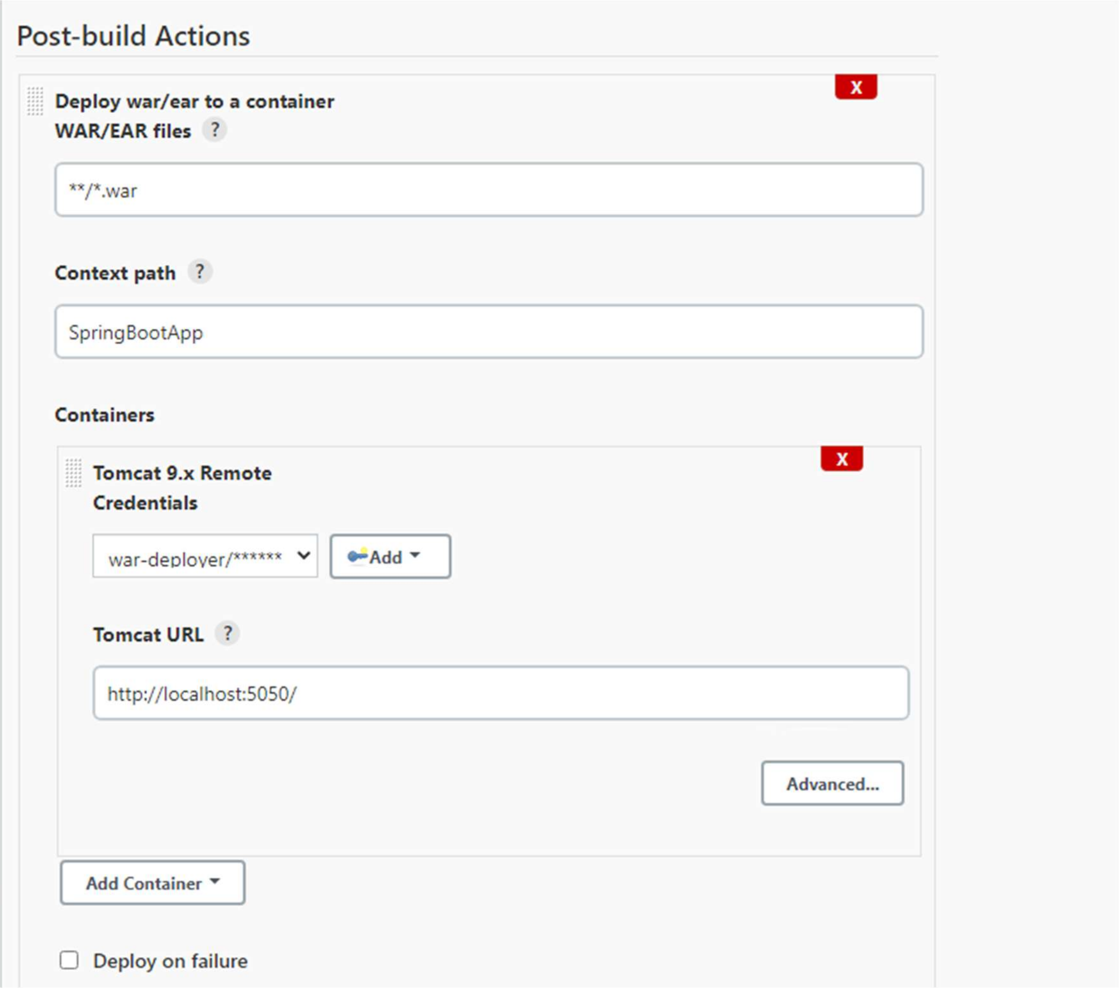
Build

Invoke top-level Maven targets ?

Maven Version: mvn

Goals: clean compile package

Post-build Actions



Post-build Actions

Deploy war/ear to a container

WAR/EAR files: **/*.war

Context path: SpringBootApp

Containers

Tomcat 9.x Remote

Credentials: war-deployer/******

Tomcat URL: http://localhost:5050/

☐ Deploy on failure

The above Jenkins Freestyle pipeline was created that can deploy the war file to the Tomcat server. The source code is fetched from github and it is set to SCM poll so that any commits in the github will be immediately built and deployed. Jenkins pipeline builds the maven project using “clean compile package” commands the the WAR file is deployed. It is deployed in Tomcat webapps and can be accessed using http://ec2-18-220-233-163.us-east-2.compute.amazonaws.com:5050/SpringBootApp/.