–

Step-by-step walkthrough for configuring Allure Report for Jenkins

**Jenkins scripted pipeline**

PREPARED BY:

PREPARED FOR:

**Version:** 1.0  
**Department:** Architect Department  
**Delivery:** End of Month

Table of Contents

[**1.** Executive Summary 3](#_Toc209554065)

[2. Create a Maven Project 3](#_Toc209554066)

[Step 1: 3](#_Toc209554067)

[Step 2: 4](#_Toc209554068)

[Step 4: 9](#_Toc209554069)

[Step 5: 9](#_Toc209554070)

[**2.** **Run Maven from project root** 10](#_Toc209554071)

[3. Configure Allure Plugin in Jenkins 13](#_Toc209554072)

[Step 1: 13](#_Toc209554073)

[Step 2: 14](#_Toc209554074)

[Step 3Allure: 14](#_Toc209554075)

[4. Configure New Job 16](#_Toc209554076)

[Step 1: 16](#_Toc209554077)

# Executive Summary

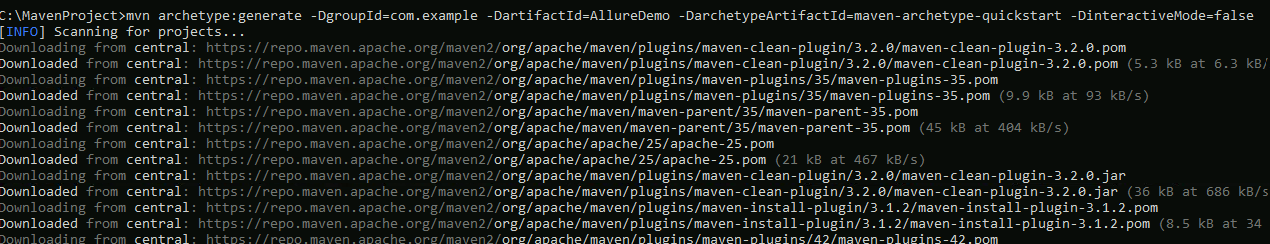
**This document provides a comprehensive guide to setting up and executing parallel builds in a development environment. It includes detailed steps, annotated screenshots, and explanations to help developers streamline their build processes and improve efficiency.**

# Create a Maven Project

## Step 1:

* Open a terminal and navigate to the folder where you want the project.
* Run:

mvn archetype:generate -DgroupId=com.example -DartifactId=AllureDemo -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false



A screen shot of a computer

AI-generated content may be incorrect.

1. **his will create a folder called AllureDemo with a basic Maven project structure:**

AllureDemo/

├─ pom.xml

├─ src/

│ ├─ main/java/com/example/App.java

│ └─ test/java/com/example/AppTest.java

A screen shot of a computer

AI-generated content may be incorrect.

Step 2: **Add Allure Dependency**

**Open pom.xml and add Allure TestNG (or JUnit) dependency under <dependencies>:**

<project xmlns="http://maven.apache.org/POM/4.0.0"

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

         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0

                             http://maven.apache.org/maven-v4\_0\_0.xsd">

  <modelVersion>4.0.0</modelVersion>

  <groupId>com.example</groupId>

  <artifactId>AllureDemo</artifactId>

  <packaging>jar</packaging>

  <version>1.0-SNAPSHOT</version>

  <name>AllureDemo</name>

  <url>http://maven.apache.org</url>

  <dependencies>

    <!-- TestNG -->

    <dependency>

      <groupId>org.testng</groupId>

      <artifactId>testng</artifactId>

      <version>7.7.0</version>

      <scope>test</scope>

    </dependency>

    <!-- Allure TestNG -->

    <dependency>

      <groupId>io.qameta.allure</groupId>

      <artifactId>allure-testng</artifactId>

      <version>2.22.0</version>

    </dependency>

  </dependencies>

  <build>

    <plugins>

      <!-- Maven Compiler Plugin -->

      <plugin>

        <artifactId>maven-compiler-plugin</artifactId>

        <version>3.10.1</version>

        <configuration>

          <source>1.8</source>

          <target>1.8</target>

        </configuration>

      </plugin>

      <!-- Maven Surefire Plugin with Allure listener -->

      <plugin>

        <groupId>org.apache.maven.plugins</groupId>

        <artifactId>maven-surefire-plugin</artifactId>

        <version>3.1.2</version>

        <configuration>

          <suiteXmlFiles>

            <suiteXmlFile>src/test/resources/testing.xml</suiteXmlFile>

          </suiteXmlFiles>

          <properties>

            <property>

              <name>listener</name>

              <value>io.qameta.allure.testng.AllureTestNg</value>

            </property>

          </properties>

        </configuration>

      </plugin>

      <!-- Allure Maven Plugin -->

      <plugin>

        <groupId>io.qameta.allure</groupId>

        <artifactId>allure-maven</artifactId>

        <version>2.11.2</version>

        <executions>

          <execution>

            <id>allure-report</id>

            <phase>verify</phase>

            <goals>

              <goal>report</goal>

            </goals>

          </execution>

        </executions>

      </plugin>

    </plugins>

  </build>

</project>

Step 3: **Add a Simple Test**

**Replace the contents of src/test/java/com/example/AppTest.java with:**

package com.example;

import org.testng.annotations.Test;

import io.qameta.allure.Description;

import static org.testng.Assert.assertEquals;

public class AppTest {

@Test

@Description("Simple test to check addition")

public void testAddition() {

int a = 5;

int b = 10;

int sum = a + b;

assertEquals(sum, 15, "Sum should be 15");

}

}

**A screen shot of a computer program

AI-generated content may be incorrect.**

Step 4: **Create a clean testng.xml in root folder**

1. **Open Notepad++ (not Word or Notepad).**
2. **Paste this exact content:**

<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd">

<suite name="AllureDemoSuite">

    <listeners>

        <listener class-name="io.qameta.allure.testng.AllureTestNg"/>

    </listeners>

    <test name="AllureDemoTests">

        <packages>

            <package name="com.example"/>

        </packages>

    </test>

</suite>

Step 4: **Save testng.xml As:**

* **Filename: testng.xml**
* **Save type: All Files (\*.\*)**
* **Encoding: UTF-8**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**Ensure correct Path as :**

**C:\MavenProject\AllureDemo\src\test\resources\testng.xml**

Step 5: **Test XML for any trailing spaces :**

* **Run Below command in path where you have save xml and ensure to trailing spaces**

**type testing.xml**

**A screen shot of a computer

AI-generated content may be incorrect.**

# **Run Maven from project root**

Open **Command Prompt** in your project root (C:\MavenProject\AllureDemo) and run:

**mvn clean test**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**A screenshot of a computer program

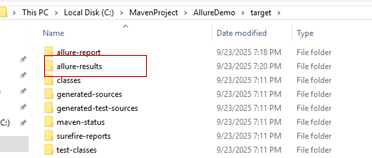
AI-generated content may be incorrect.**

* Now you should report in json inside **allure-result** under Root

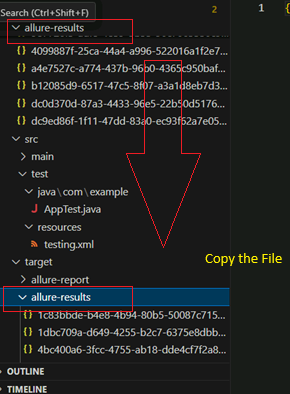
A screen shot of a computer

AI-generated content may be incorrect.

* Create the folder **allure-results** under **taget**



* Copy the JSON file from **AllureDemo** root into **target\allure-results:**

****

**Step 2: Generate the report**

**allure serve target/allure-results**

**A screen shot of a computer code

AI-generated content may be incorrect.**

**A screenshot of a report

AI-generated content may be incorrect.**

**Push your Code to git hub**

**A computer screen shot of a computer code

AI-generated content may be incorrect.**

[**https://github.com/bipeensinha/allure-jenkin**](https://github.com/bipeensinha/allure-jenkin)

**A screenshot of a computer

AI-generated content may be incorrect.**

# Configure Allure Plugin in Jenkins

Step 1: **Install Jenkins Allure Plugin**

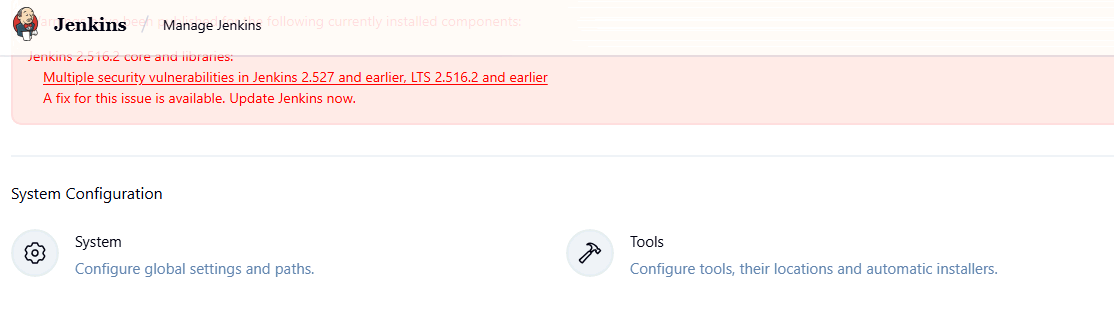
1. Open Jenkins → **Manage Jenkins** → **Manage Plugins**.
2. Go to **Available Plugins**, search for **Allure Jenkins Plugin**.
3. Install it and restart Jenkins if required.

A screenshot of a chat

AI-generated content may be incorrect.

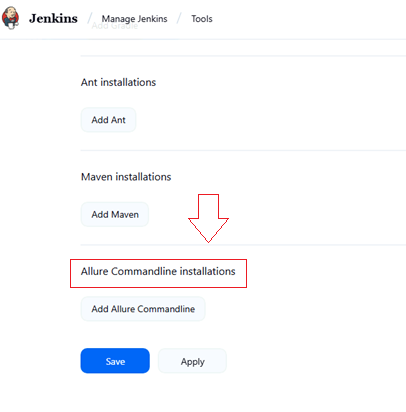
Step 2: **Go to Global Tool Configuration**

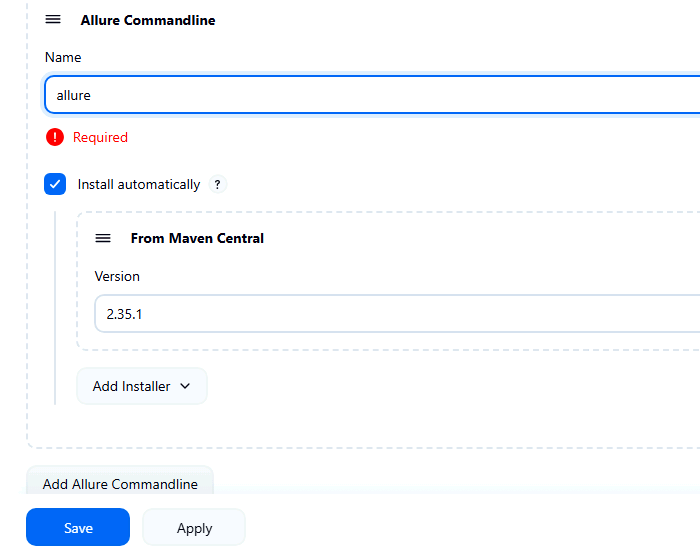
* Navigate to: **Manage Jenkins → Global Tool Configuration**
* Scroll down — you should see **Allure Commandline** there.



Step 3Allure: **Add Allure Commandline Tool**

1. Under **Allure Commandline**, click **Add Allure Commandline**
2. Name: allure
3. Check **Install automatically** (Jenkins will download the latest Allure binary for you)
4. Save

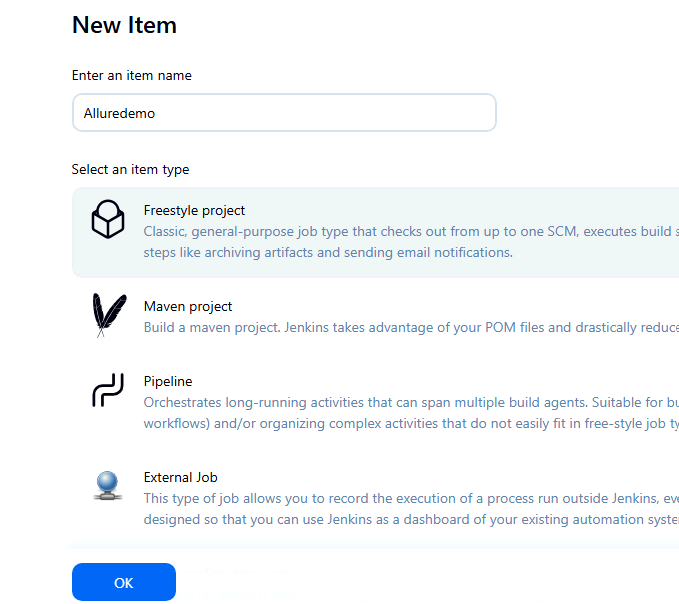




# Configure New Job

Step 1: **Create Jenkins Job (Freestyle)**

1. **Create New Item → *Freestyle Project***

****

1. **Under Source Code Management:**
   * **Select Git**
   * **Add the repo URL (e.g.,** [**https://github.com/yourusername/AllureDemo.git**](https://github.com/yourusername/AllureDemo.git)**)**

**A screenshot of a computer

AI-generated content may be incorrect.**

1. **Under Build → *Invoke top-level Maven targets***

**A screenshot of a computer

AI-generated content may be incorrect.**

**clean test**

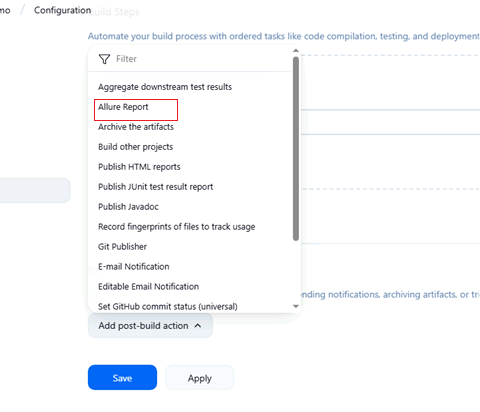
**This runs your tests and produces target/allure-results folder inside the Jenkins workspace.**

**A screenshot of a computer program

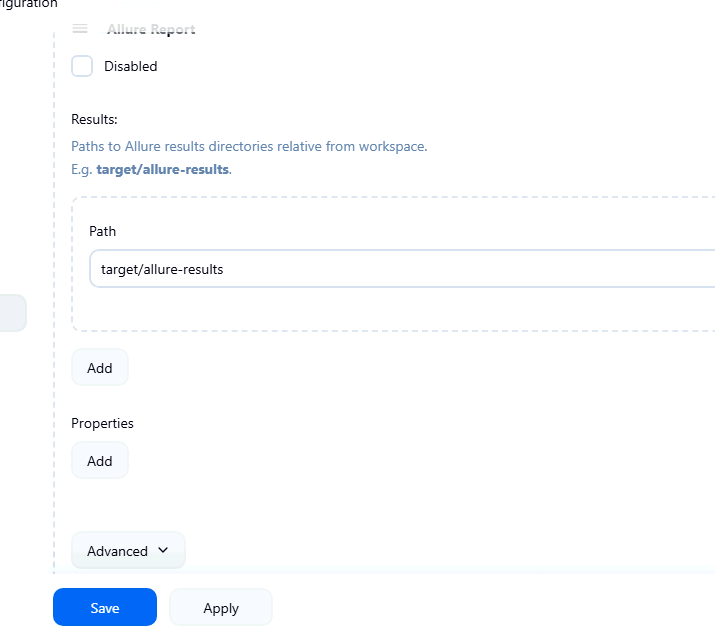
AI-generated content may be incorrect.**

1. **Then: Allure Report in Jenkins**

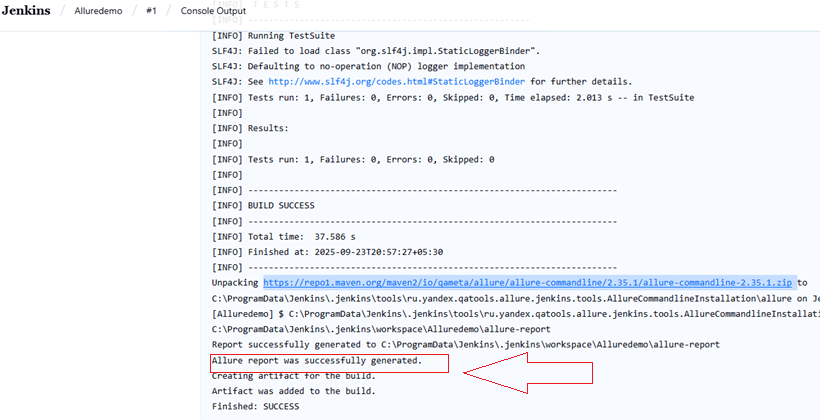
**Post-build Action → Allure Report**:



* Results path: target/allure-results

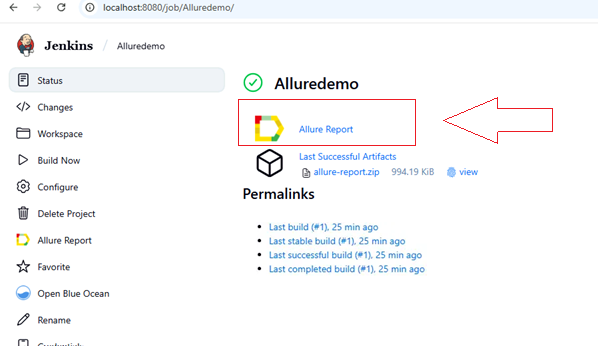


* Version: Allure-2.22
* After the job runs, you’ll see an **Allure Report link** in Jenkins job re
* Save → **Build Now**



**Step 2: View Report**

1. Once the build finishes, go to **Build Page → Allure Report**.



1. You’ll see:
   * Test statistics (passed, failed, skipped)
   * Trend graphs
   * Detailed test results
   * History (if enabled)

