Configuring Jenkins for Appium Test Execution

To configure Jenkins for Appium test execution, follow these steps to set up a robust CI/CD pipeline:

**1. Install Jenkins**

* Download and install Jenkins from the [official Jenkins site](https://www.jenkins.io/).
* Start Jenkins and complete the initial setup process.

**2. Install Required Plugins**

Go to Manage Jenkins > Manage Plugins, and install the following:

* **Maven Integration Plugin**
* **JUnit Plugin**
* **HTML Publisher Plugin**
* **Pipeline Plugin**
* **Git Plugin** (optional for version control)

**3. Configure Global Environment Variables**

Go to Manage Jenkins > Configure System, then:

* **JAVA\_HOME**: Path to JDK (e.g., /usr/lib/jvm/java-11-openjdk)
* **ANDROID\_HOME**: Path to Android SDK (e.g., /path/to/android/sdk)
* **APPIUM\_HOME**: Path to Appium installation (e.g., /usr/local/lib/node\_modules/appium)

Ensure the paths to tools like adb, gradle, and appium are added to PATH.

**4. Configure Node and Appium Setup**

* Install **Node.js** and **Appium** globally if not already done:

bash

CopyEdit

npm install -g appium

npm install -g appium-doctor

* Verify setup using:

bash

CopyEdit

appium-doctor

**5. Create Appium Test Project**

Ensure the Appium project has:

* Maven configuration (pom.xml) with dependencies for Appium and TestNG/JUnit.
* Appium test scripts defined properly.

**Sample pom.xml:**

xml

CopyEdit

<dependencies>

<dependency>

<groupId>io.appium</groupId>

<artifactId>java-client</artifactId>

<version>8.3.0</version>

</dependency>

<dependency>

<groupId>org.testng</groupId>

<artifactId>testng</artifactId>

<version>7.7.0</version>

</dependency>

</dependencies>

**6. Create Jenkins Pipeline Job**

1. Go to New Item > Pipeline.
2. Enter a name and select Pipeline.
3. Under Pipeline Configuration, select Pipeline Script.

**7. Define Jenkins Pipeline**

Here’s a sample Jenkinsfile for Appium test execution:

groovy

CopyEdit

pipeline {

agent any

environment {

JAVA\_HOME = "/path/to/jdk"

ANDROID\_HOME = "/path/to/android/sdk"

PATH = "${env.PATH}:${env.JAVA\_HOME}/bin:${env.ANDROID\_HOME}/platform-tools"

}

stages {

stage('Checkout Code') {

steps {

git url: 'https://github.com/your-appium-project.git'

}

}

stage('Install Dependencies') {

steps {

sh 'npm install -g appium'

}

}

stage('Execute Tests') {

steps {

sh './mvnw clean test'

}

}

}

post {

always {

junit '\*\*/target/surefire-reports/\*.xml'

archiveArtifacts artifacts: '\*\*/target/screenshots/\*.png', allowEmptyArchive: true

}

}

}

**8. Trigger Test Execution**

* Build the pipeline manually or schedule it for automated execution.

**9. View Test Reports**

* Check test results under Test Results and view artifacts like screenshots under Artifacts.

**10. (Optional) Set Up Parallel Execution**

To enhance efficiency, you can execute tests on multiple devices using Appium Grid in Jenkins.