L’installazione di Selenium nel server è fondamentale per far andare i test nel server.

Per fare ciò andiamo ad utilizzare Maven, che scaricherà tutti i collegamenti Java e tutte le sue dipendenze utilizzando il file maven pom.  
Il primo passo del processo per l’installazione di Selenim è quindi, quello di creare una cartella che conterrà i file del progetto Selnium, per poi creare il file pom.xml, che conterrà le configurazioni del progetto.

<? xml version="1.0" encoding="UTF-8"?>

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

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

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

<modelVersion>4.0.0</modelVersion>

<groupId>SeleniumTest</groupId>

<artifactId>SeleniumTest</artifactId>

<version>1.0</version>

<dependencies>

<dependency>

<groupId> org.seleniumhq.selenium </groupId>

<artifactId>selenium-server</artifactId>

<version>3.0.1</version>

</dependency>

<dependency>

<groupId> org.seleniumhq.selenium </groupId>

<artifactId>htmlunit-driver</artifactId>

<version>2.34.0</version>

</dependency>

<dependency>

<groupId> org.junit.jupiter</groupId>

<artifactId>junit-jupiter</artifactId>

<version>5.5.0-M1</version>

</dependency>

</dependencies>

</project>

Dopo di ciò si usa il comando ”mvn clean install” dalla riga di comando dentro la directory del progetto, questo scaricherà Selenium, tutte le sue dipendenze e le aggiungerà al progetto.

Il passo successivo è quello di importare il progetto maven in IntelJ e creare i test junit5(jupiter).

A questo punto vengono installate tutte le librerie nel server, insieme a firefox e geckodriver:

xvfb

libxi6

libgconf-2-4

default-jdk

firefox

geckodriver-v0.24.0-linux64

apiguardian-api-1.0.0.jar

junit-jupiter-api-5.4.2.jar

junit-platform-commons-1.4.2.jar

opentest4j-1.1.1.jar

selenium-server-standalone-3.13.0.jar

testng-6.8.7.jar.zip