Download Jenkins from the URL <http://mirrors.jenkins.io/windows/latest>

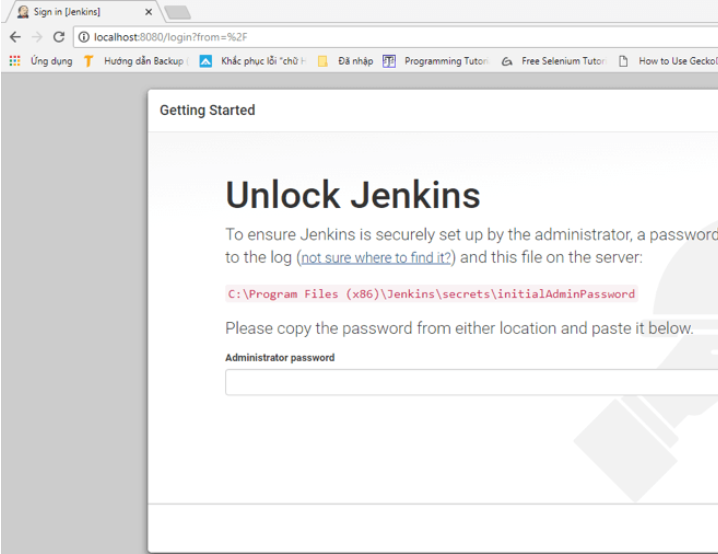
Jenkins url

http://localhost:8080

username: admin

password: (initial password path: C:\Program Files (x86)\Jenkins\secrets\InitialAdminPassword)

Open URL [http://localhost:8080](http://localhost:8080/)

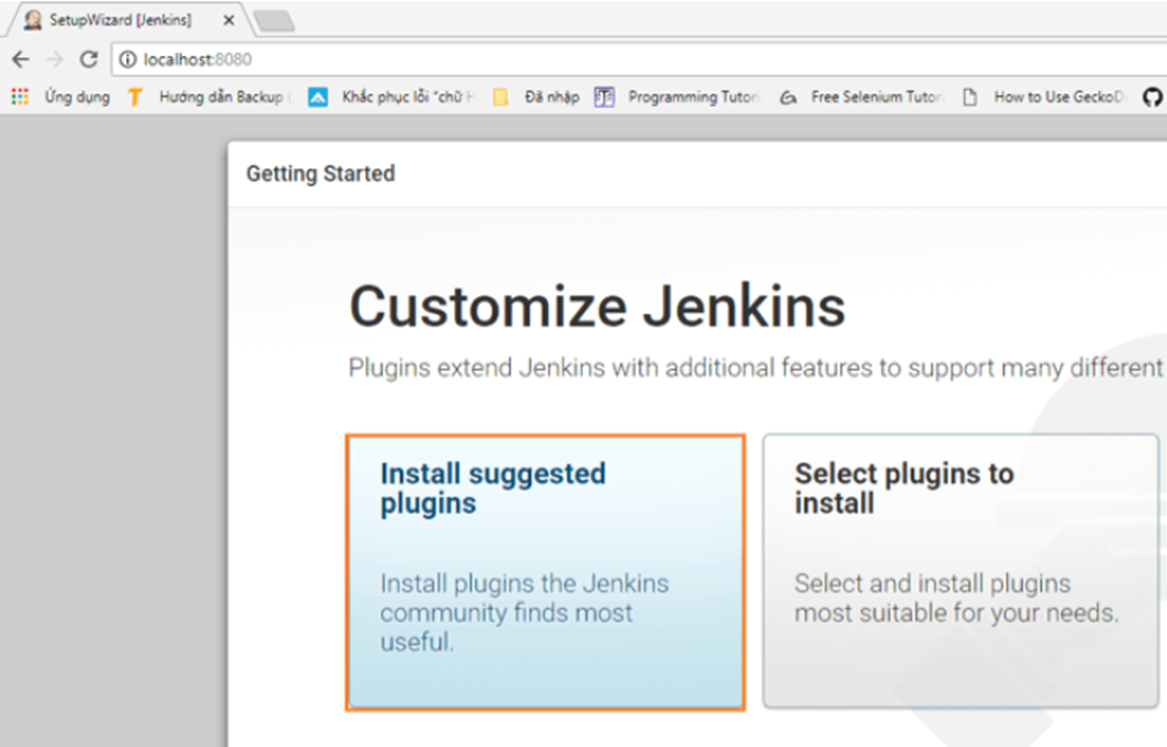


Incase Jenkins doesn’t get started, use below command to start it:

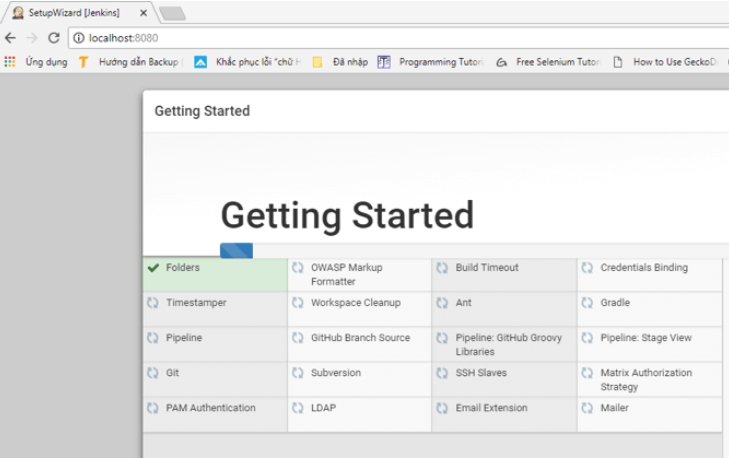
C:\Program Files (x86)\Jenkins>java -jar jenkins.war --httpPort=9090

and wait until u see “ Jenkins is fully up and running”

To unlock Jenkins, copy the password from the file at**C:\Program Files (x86)\Jenkins\secrets\initialAdminPassword** and paste it in the **Administratorpassword** field. Then, click the **"Continue"** button



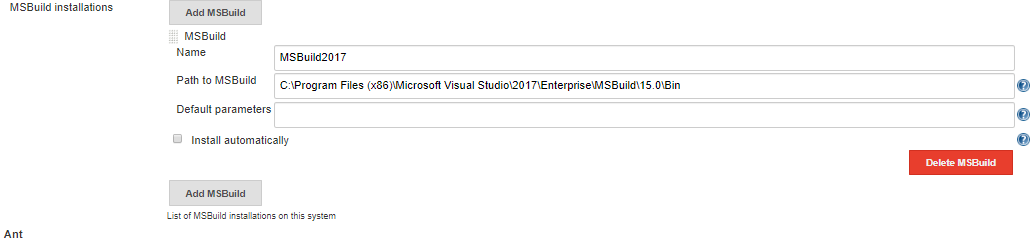
In Next Screen, wait until all plugins are installed

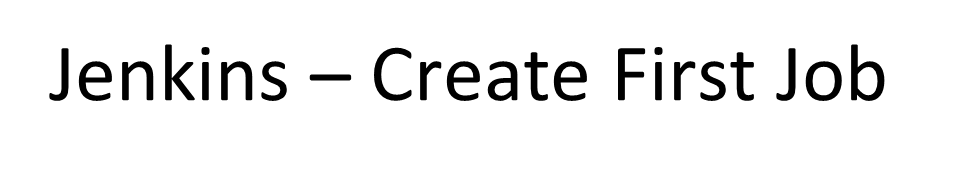


Note: Install the MsBuild Plugin in order to run the Visual Studio Project or Solution File.

After Install the MsBuild Plugin then configure the Mubuild Path.

Msbuild path: C:\Program Files (x86)\Microsoft Visual Studio\2017\Enterprise\MSBuild\15.0\Bin

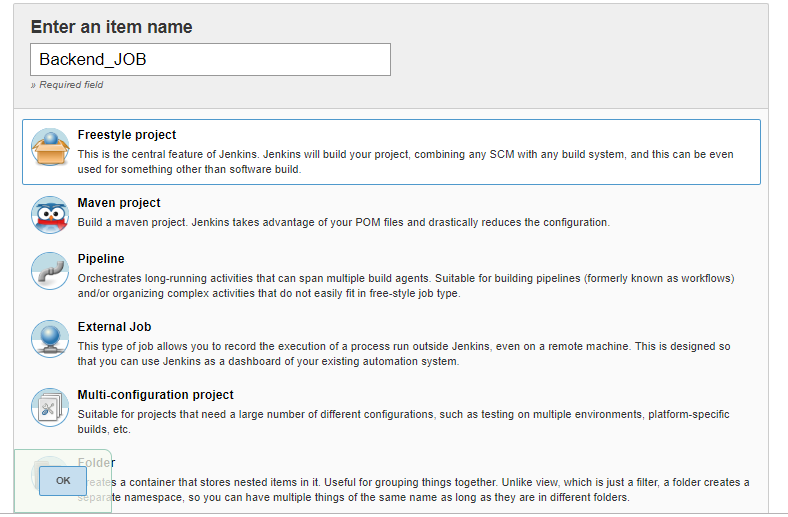




After Login to Jenkins, click New Item

Provide a Name for the New Job, select “Freestyle”

Click Ok.

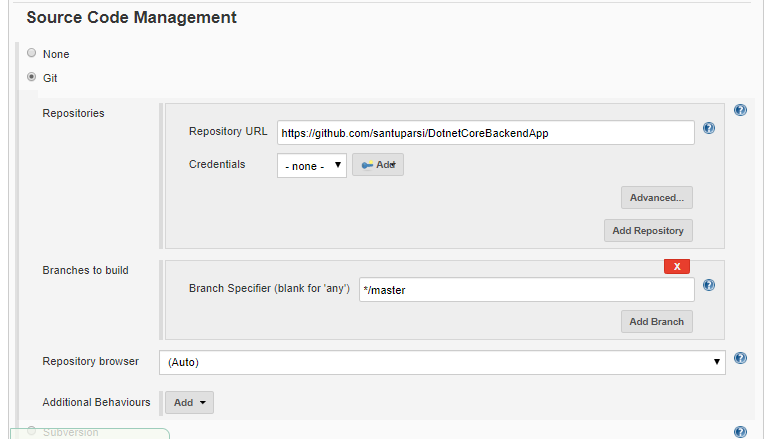


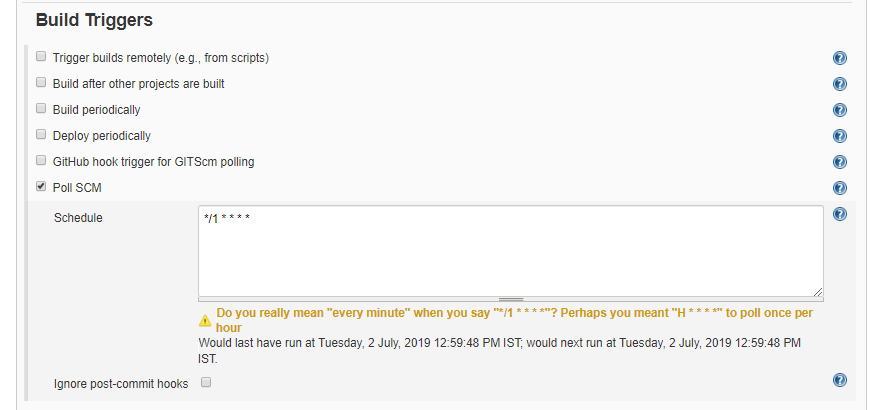
Now on the next web page, under “Source Code management” select “Git”

This option gets expanded, and asks for GIT repo URL

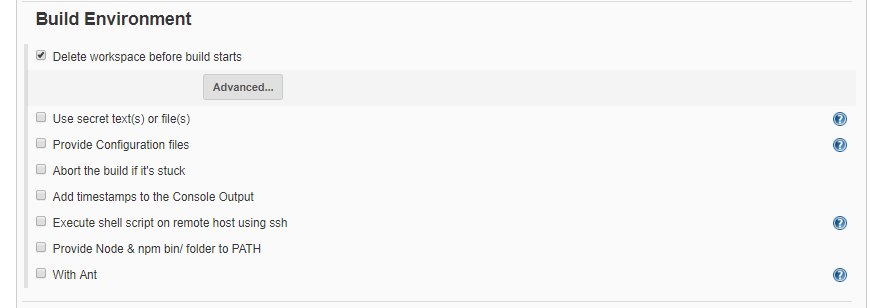
In Build Trigger, select Poll SCM, specify \* \* \* \* \*

Above indicates that SCM need to be polled for every one minute, for changes. Once Jenkins Job finds changes, build is initiated.





Next Under Build Environment select Delete workspace before build starts

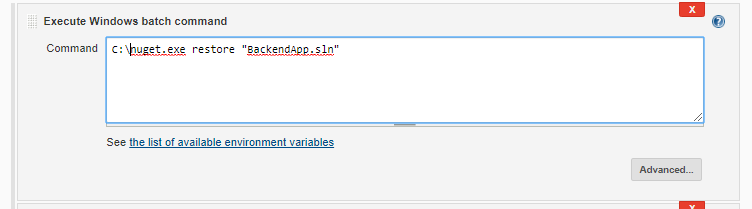


Next Step: Restore the Nuget Packages

**Select Windows Batch Command under Build Section and enter command**

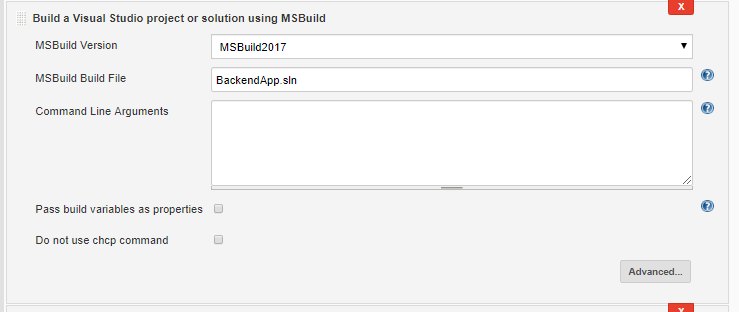
**C:\nuget.exe restore "BackendApp.sln"**

**Note: download the nuget.exe and place it in C Drive.**



Next Step: Build the project using MSBuild

**Select Build a Visual Studio Project or Solution using MSBuild Command**



Next Step: Run the Unit test cases using Nunit

Note: Install the Nunit Plug in before run the below command.

**Crate Windows Batch Command under Build Section and enter command**

**"D:\CTS\Batch10\GIT\HandsOnNunit\packages\NUnit.ConsoleRunner.3.9.0\tools\nunit3-console.exe" D:\CTS\Batch10\GIT\HandsOnNunit\TestLib\bin\Debug\TestLib.dll**



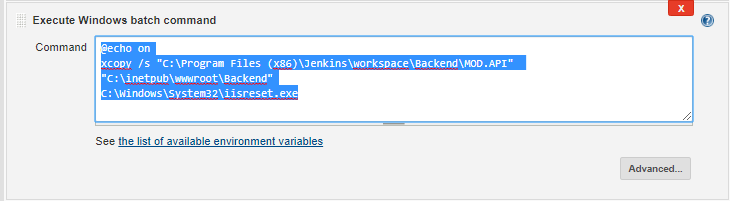
Next Step: Deploy the Backend App into IIS

Crate Windows Batch Command under Build Section and enter command

@echo on

xcopy /s "C:\Program Files (x86)\Jenkins\workspace\Backend\MOD.API" "C:\inetpub\wwwroot\Backend"

C:\Windows\System32\iisreset.exe



Click the Save button Job will create successfully.

Finally Click the Buildnow button

Windows Batch commands are executed in a sequence order and finally will get the Success Message.

Console Output

