|  |  |
| --- | --- |
| D:\PAL\Logo_4_21_15\Logo_4_21_15\Primary Logo\png_files\EPAM_LOGO_Full_Color_RGB.png | CDP Automated Testing Mentoring Program  2017-Q2 |

HOME TASK Module 2.2: dive into ci with jenkins

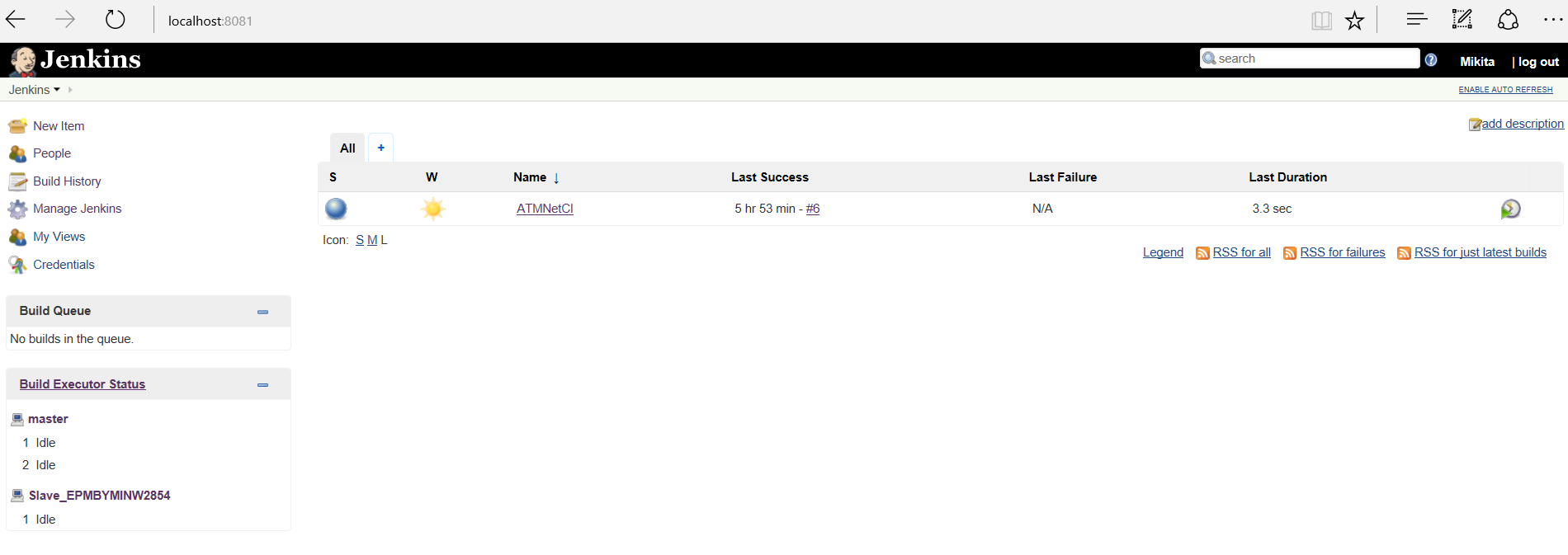
## home task

As a basis for this home task, use this git repository: <https://github.com/vitalliuss/helloci>

1. Jenkins Installation:

* Install Jenkins server on local or remote machine;
* Change Jenkins default port to 8081;
* Create slave node on remote or local machine (you may use the same (one) machine for master and slave);

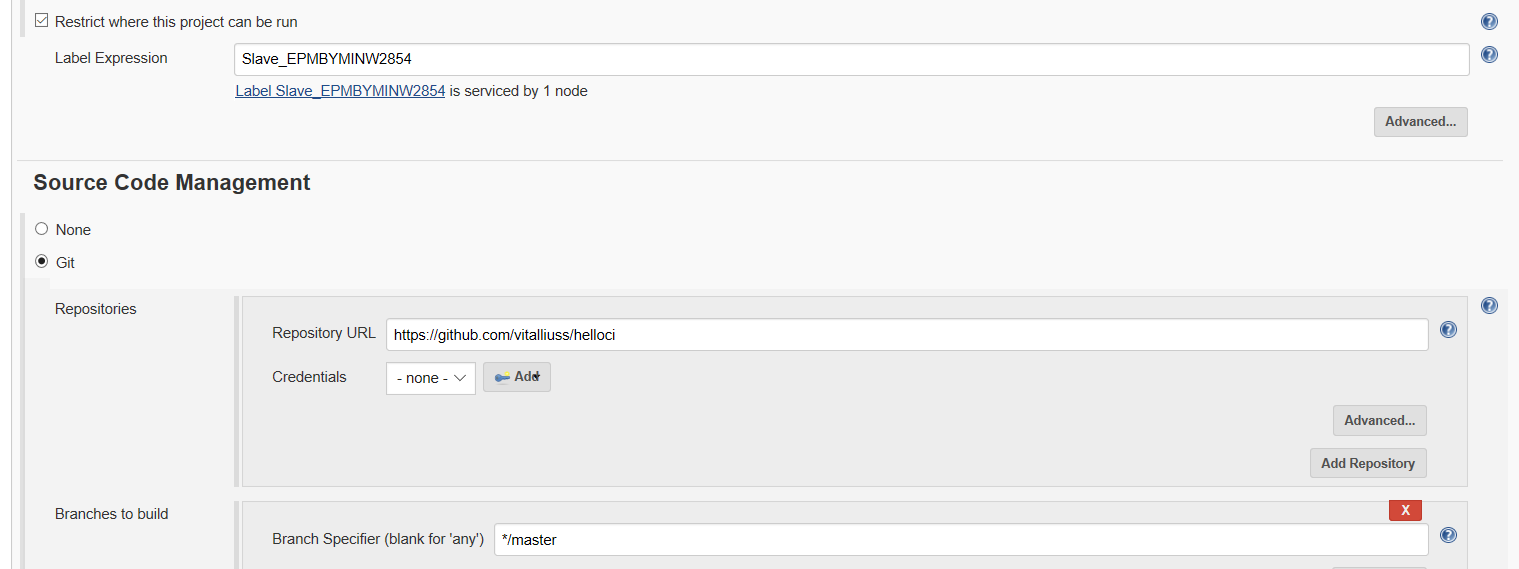
MD: Installed on local machine, changed port and slave node created on the same machine using JNLP agent.



1. Job configuration:

* Get code from git repository: <https://github.com/vitalliuss/helloci>
* JAVA: build and run unit tests from Java folder. Use “test” goal for maven

.NET: build solution (\*.sln) with MSBuild



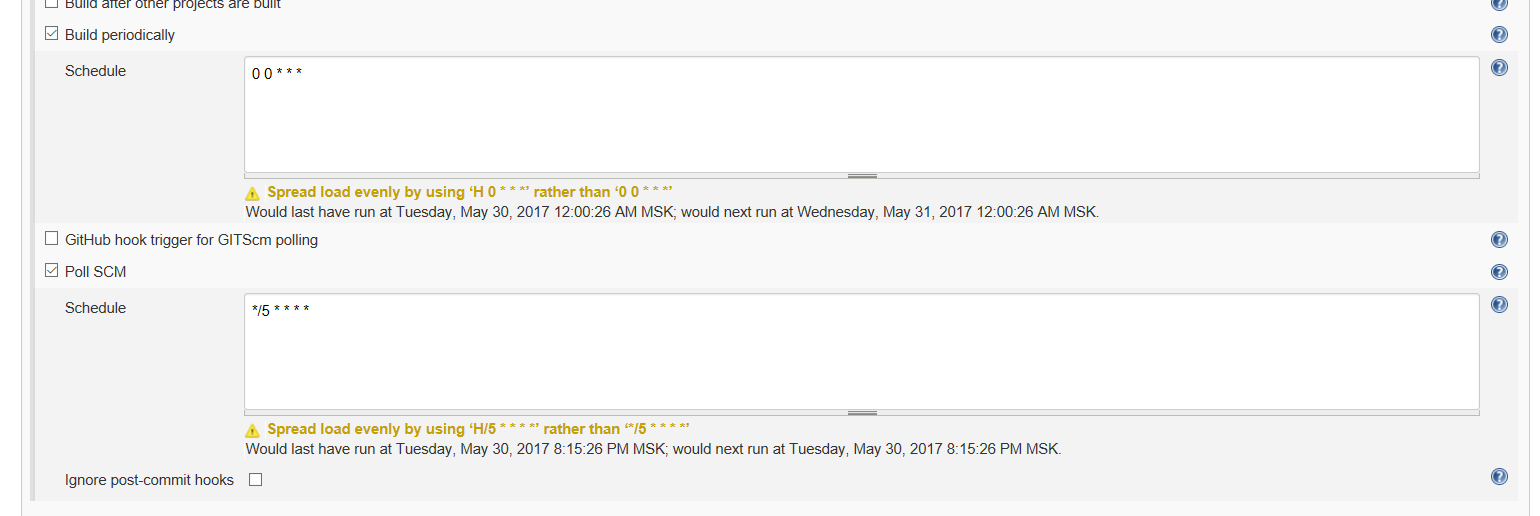
MD: Solution from GIT is successfully being build using MSBuild(see log below)

JS: build helloworld.js from JS folder

* Set build triggers:

1. After commit in VCS no longer than in 5 minutes
2. At midnight every day

MD: Triggers setup



* Job should run on slave only

MD: Console output on the latest build

Started by user [Mikita](http://localhost:8081/user/MikitaD)

Building remotely on [Slave\_EPMBYMINW2854](http://localhost:8081/computer/Slave_EPMBYMINW2854) (Slave) in workspace C:\Slave\workspace\ATMNetCI

> C:\Program Files (x86)\Git\bin\git.exe rev-parse --is-inside-work-tree # timeout=10

Fetching changes from the remote Git repository

> C:\Program Files (x86)\Git\bin\git.exe config remote.origin.url <https://github.com/vitalliuss/helloci> # timeout=10

Fetching upstream changes from <https://github.com/vitalliuss/helloci>

> C:\Program Files (x86)\Git\bin\git.exe --version # timeout=10

> C:\Program Files (x86)\Git\bin\git.exe fetch --tags --progress <https://github.com/vitalliuss/helloci> +refs/heads/\*:refs/remotes/origin/\*

> C:\Program Files (x86)\Git\bin\git.exe rev-parse "refs/remotes/origin/master^{commit}" # timeout=10

> C:\Program Files (x86)\Git\bin\git.exe rev-parse "refs/remotes/origin/origin/master^{commit}" # timeout=10

Checking out Revision 1825aa42a6e6ce077626096fd7c00dd17fc37eed (refs/remotes/origin/master)

> C:\Program Files (x86)\Git\bin\git.exe config core.sparsecheckout # timeout=10

> C:\Program Files (x86)\Git\bin\git.exe checkout -f 1825aa42a6e6ce077626096fd7c00dd17fc37eed

> C:\Program Files (x86)\Git\bin\git.exe rev-list 1825aa42a6e6ce077626096fd7c00dd17fc37eed # timeout=10

Path To MSBuild.exe: C:\Program Files (x86)\MSBuild\12.0\Bin\MSBuild.exe

Executing the command cmd.exe /C " chcp 1252 && "C:\Program Files (x86)\MSBuild\12.0\Bin\MSBuild.exe" .NET/Calculator.sln " && exit %%ERRORLEVEL%% from C:\Slave\workspace\ATMNetCI

[ATMNetCI] $ cmd.exe /C " chcp 1252 && "C:\Program Files (x86)\MSBuild\12.0\Bin\MSBuild.exe" .NET/Calculator.sln " && exit %%ERRORLEVEL%%

Active code page: 1252

Microsoft (R) Build Engine version 12.0.21005.1

[Microsoft .NET Framework, version 4.0.30319.42000]

Copyright (C) Microsoft Corporation. All rights reserved.

Building the projects in this solution one at a time. To enable parallel build, please add the "/m" switch.

Build started 5/30/2017 1:52:52 PM.

Project "C:\Slave\workspace\ATMNetCI\.NET\Calculator.sln" on node 1 (default targets).

ValidateSolutionConfiguration:

Building solution configuration "Debug|Mixed Platforms".

Project "C:\Slave\workspace\ATMNetCI\.NET\Calculator.sln" (1) is building "C:\Slave\workspace\ATMNetCI\.NET\Calculator\Calculator.csproj" (2) on node 1 (default targets).

ResolveAssemblyReferences:

A TargetFramework profile exclusion list will be generated.

GenerateTargetFrameworkMonikerAttribute:

Skipping target "GenerateTargetFrameworkMonikerAttribute" because all output files are up-to-date with respect to the input files.

CoreCompile:

Skipping target "CoreCompile" because all output files are up-to-date with respect to the input files.

CopyFilesToOutputDirectory:

Calculator -> C:\Slave\workspace\ATMNetCI\.NET\Calculator\bin\Debug\Calculator.dll

Done Building Project "C:\Slave\workspace\ATMNetCI\.NET\Calculator\Calculator.csproj" (default targets).

Project "C:\Slave\workspace\ATMNetCI\.NET\Calculator.sln" (1) is building "C:\Slave\workspace\ATMNetCI\.NET\CalculatorTest\CalculatorTest.csproj.metaproj" (3) on node 1 (default targets).

Project "C:\Slave\workspace\ATMNetCI\.NET\CalculatorTest\CalculatorTest.csproj.metaproj" (3) is building "C:\Slave\workspace\ATMNetCI\.NET\CalculatorTest\CalculatorTest.csproj" (4) on node 1 (default targets).

GenerateTargetFrameworkMonikerAttribute:

Skipping target "GenerateTargetFrameworkMonikerAttribute" because all output files are up-to-date with respect to the input files.

CoreCompile:

Skipping target "CoreCompile" because all output files are up-to-date with respect to the input files.

\_CopyAppConfigFile:

Skipping target "\_CopyAppConfigFile" because all output files are up-to-date with respect to the input files.

CopyFilesToOutputDirectory:

CalculatorTest -> C:\Slave\workspace\ATMNetCI\.NET\CalculatorTest\bin\Debug\CalculatorTest.dll

Done Building Project "C:\Slave\workspace\ATMNetCI\.NET\CalculatorTest\CalculatorTest.csproj" (default targets).

Done Building Project "C:\Slave\workspace\ATMNetCI\.NET\CalculatorTest\CalculatorTest.csproj.metaproj" (default targets).

Done Building Project "C:\Slave\workspace\ATMNetCI\.NET\Calculator.sln" (default targets).

Build succeeded.

0 Warning(s)

0 Error(s)

Time Elapsed 00:00:00.17

[ATMNetCI] $ cmd /c call C:\Users\MIKITA~1\AppData\Local\Temp\jenkins2226805335857800755.bat

C:\Slave\workspace\ATMNetCI>ECHO Content 1>output.txt

C:\Slave\workspace\ATMNetCI>exit 0

Archiving artifacts

Finished: SUCCESS

As the result share the link to Jenkins OR set of detailed screenshots of Jenkins and job configuration & logs with your mentor.

## acceptance criteria

1. Jenkins server has been set up on local or remote machine with default port 8081
2. Slave node has been created
3. Job has been configured with necessary settings and build actions
4. Job is triggered according to task
5. Job has been set up to run on slave node only

## BONUS TASK

1. Make server on linux (use Cloud or virtual machine)
2. Make node on linux
3. Configure job history tracking and server backup