You have been assigned on a project as a build engineer. Project is a solution for embedded systems. Your team members are strong C++ developers who are focused on development of the solution. At the moment they are having difficulties with integration of developed modules. Team has to waste a lot of time on manual activities like checking code style, building, testing and so on. Team decided to try the Continuous Integration (**CI**) approach and to automate some everyday activities. Your team isn't familiar with CI and they have asked you to prepare proposal which should include documentation and minimum viable product (**MVP**).

Work Description

* Put your documentation into README.md file. It should be in *markdown* format;

**Tasks**

**1. You have to prepare development environment which will be based on docker technology and include the following instances:**

* Jenkins master;
* Jenkins Agent;
* Artifactory;

**2. Automate build process for a project.**

* Find C/C++ project at GitHub (check a license of a project before you start using one). For example, you can take the [curl](https://github.com/curl/curl);
* Implement **scripted** Groovy pipeline for you project. This pipeline should contain the following steps:
  + checkout project repository
  + build
  + execute unit tests
  + prepare build artifact
  + push the artifact into Artifactory;
  + build results and a link to Artifactory should be available on build dashboard in Jenkins.

**3. Tests for Groovy pipeline**

* With [JenkinsPipelineUnit](https://github.com/jenkinsci/JenkinsPipelineUnit" \t "_blank) framework write tests for pipeline:
  + Successful execution test (load pipeline, print call-stack)
  + Stages test: test, that all expected stages were executed
  + Artifactory push tests: check that call to Artifactory was executed with expected arguments.

**4. Configure pipeline repo Quality Gates**

* Prepare Gradle file which allows you to run CodeNarc checks with default set of rules against your pipeline script and tests prepared at previous step.
* Configure a quality gate job that runs both checks for each Pull/Merge request to your repository publishing the results at build page.

**5. Prepare documentation.**

All documentation for your project has to be put into README.md file and written in markdown.

**Acceptance criteria:**

* File of docker-compose is prepared and located in git repository;
* The environment is launched via docker-compose. Environment is available and doesn't require additional manual configuration. If so, the necessary steps are provided in documentation;
* Pipeline job takes project from GitHub/GitLab/EPAM and builds, tests, checks and etc.;
* There's a binary artifact available in Artifactory;
* Build results and link to Artifactory are available on build dashboard in Jenkins;
* Pipeline Quality Gates are configured for Pull/Merge requests to your repository;
* The results for pipeline checks provided to Pull/Merge request;
* Documentation is available.