**CI CD Homework**

You’ve just joined the project of your dream, to the most desired customer. Looks like it’s a great chance for your career growth. If this project is successful, we can figure on further cooperation and attracting initiatives. Congratulations!

It seems that CI/CD was not configured for the testing framework yet, and the DevOps team is very busy with pipelines migration, so you will have to setup it by your own.

Your main task now is to set up the code repository, version control and CI/CD infrastructure as soon as possible.

Some project info:

|  |  |
| --- | --- |
| Delivered service | Financial data processing & Billing system |
| Code Language | Python |
| Team setup | 5 Dev; 1 DQ, 3 DevOps |
| Architecture preferences: | Docker, Jenkins, Git |
| Service preferences | Open Source |
| Development approach | Agile |
| Release frequency | 1 per 2 weeks in pre-prod; 1 a month - prod |

What are you going to do about it?

Here are some tips:

1. Using Python scripts from Test Automation module and your repository, set it up for command use: choose a merging strategy for working with the repository, justify your choice, create sample branches and PRs according to the chosen strategy.
2. Install Docker on your computer
3. Run Docker container with Jenkins
4. Set up CI Pipeline:
   1. Set up an open repository at GitHub
   2. Configure Jenkins to access to the GitHub repository
5. Design Jenkins CI pipeline with step by step running python tests from your test file and push it into your repository.
6. Design CD Pipeline:
   1. The pipeline should copy the branch code to another (release) branch of the repository or another repository / local folder, depending on the chosen strategy and architecture

*\** *Use bash commands to display the results of pipeline steps, copy and run the files.*

Evaluation criteria

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Requirement** | **% total score** |
| Code repository | Merging strategy is chosen; the choice is justified | 10 |
| Code repo is configured according to the chosen strategy: the corresponding branches have been created; examples of commits and merges in accordance with the chosen strategy | 10 |
| Environment setup | Docker installed; deployed container with Jenkins; Configured Jenkins access to the GitHub repository | 20 |
| CI Pipeline | CI pipeline is in code repo and visible for Jenkins | 20 |
| Pipeline is runs/builds .py scripts | 10 |
| Pipeline runs .py with tests | 10 |
| Pipeline shows run status | 10 |
| CD Pipeline: | Moves source files to a separate branch / repository | 10 |