Acert Analytics

Machine Learning/MLOps Technical Challenge

Overview:-

This challenge is designed to exhibit a broad range of skill-set needed for this role as well as the type of work that you will be doing as part of this role. This challenge can be thought of as a Feature that needs to be implemented as part of the role. In case you get blocked or something is not clear please use your judgement to proceed.

You're expected to create an AWS/GCP/Azure cloud account for the challenge. The free tier should meet all the requirements of this challenge and you won't need to pay anything. In case you go over the free tier limits, the cost should be only a few dollars but you will face this situation more likely due to choices made for the solution rather than the requirements of the challenge.

There are 2 problems in this technical challenge. You are expected to do either one of these. We don't expect you to do both problems.

If you clear the technical challenge we'll go over the code/design in the subsequent F2F round. The F2F round will focus on the following:-

- What choices you made and why?
- Any trade-offs you had to make and why?

Timeline:-

We think **One Week** should be good enough to complete this challenge. Ideally, it should not take this long if you know what you are doing. However, life is unpredictable so do let us know if you need more time for completing this. (You won't be penalized if you ask for more time)

Tech stack:-

1. Minimum python 3.7 runtime
2. Deployment on a cloud platform of your choice is preferred.
3. Scikit-learn
4. Tensorflow or PyTorch (if needed)

Problem 1:-

Create an API that predicts the price of a house using the Boston house-prices dataset.
https://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_boston.html

Minimum Requirements:-

- An API/APIs that can be invoked via curl or Postman
- Input - features of a house
- Output - Price of the house
- Dockerize the solution

Model Accuracy is **NOT** important for this problem. Remember this is NOT a Data Science role. A working ML model that can predict a price is fine. We are NOT testing model building skills. We want to see whether you can design the ML lifecycle for a toy problem. If you can use an open-source model for this problem that's fine as well (but do mention the source).

Alternate Problem

Problem 2:-

Create a simple ToDo List application i.e. a list of tasks that a user wants to complete in some time.

Minimum Requirements:-

- The application should expose the following APIs:
    - Create task
    - Update task
    - Read task
    - Delete task
    - Read all tasks
    - Delete all tasks
- Dockerize the solution

Delivery:-

For the problem you solve please do the following:-

- Share the code in a repo on github/bitbucket
- Instructions on how to execute the code
- Instructions on how to test the APIs

Bonus points:-

Each of the following (not in any specific order) gives you bonus points (but these are NOT mandatory). These are valid irrespective of which problem you solve. All the following carry equal points.

- Deploy the solution using serverless architecture (e.g. AWS Lambda/Azure functions..)
- Automate infrastructure provisioning - Infrastructure as Code (e.g. terraform, chef etc...)
- Unit-testing & code coverage
- Integrating CI/CD and deployment using Jenkins/Azure Pipelines/CircleCI etc…
- Well documented code
- Modularized code (think breaking the code into functions, classes, modules etc…)
- Documentation describing any trade-offs/design choices

No bonus points for completing both the problems. We would like to see code quality so please spend time on solving either one of the problems.