1) **BACKEND**

Write a server and client implementation for a checklist service with the following requirements:

1. A user can:
    a. Create a named checklist to contain TODOs
    b. Retrieve owned checklists
    c. Add a TODO to a checklist. The user must enter a string description for the TODO. The TODO is created in “not done” status.
    d. Mark a TODO as “done” in the checklist
    e. Mark a TODO as “not done” in the checklist
    f. Delete a checklist if all contained TODOs are “done”

Use whatever high level language,for server implementation. Use whatever language for client implementations. Use whatever DB for data storage if needed. Place both server and client implementations in the same GIT repository with detailed instructions on how to test the implementations. Add to the GIT repository a documentation file with the definition and description of the REST API implemented.

The output of the exercise must be the link to the GIT repository that will be cloned, tested and evaluated.

2) **IOT**

You have an industrial machine and you need to extract data points from it with a data gathering unit (a linux based industrial pc); data must be forwarded to a relational database for storage and retrieval. You can write any code and install any package in the industrial pc.
Describe technical choices for a solution that must:

- Avoid loss of data
- Avoid duplication of data

Focus on the logic implemented by the linux pc and the database insertion queries.