Junior CV Engineer assignment

This assignment encompasses various topics and tools for evaluating how well your CV aligns with the requirements of the position. Completing all tasks is not a mandatory requirement for selection. The evaluation process will consider not only the number of tasks completed but also the quality of the solutions adopted, and the documentation created alongside them.

Task 1: Set Up the Development Environment

- **Git Setup:**
    a. Create a private GitHub repository (shared with the following GH users: pietrobalatti, jmgandarias, mleonori, 4TINI).

Task 2: Implement Functionalities:

- **OpenCV:**
    a. Write a Python script or C++ executable that uses OpenCV to perform a video capture from a USB/integrated webcam.
    b. Resize the image to a 300x300.
    c. Convert it to gray scale.
    d. Implement a face detection algorithm among the pre-trained classifiers provided by OpenCV or choose an alternative classifier of your preference.
    e. Display the video stream.
    f. Display the acquisition fps on the display window.
- **ROS:**
    g. Create a ROS 2 package containing a launch file and 2 ROS nodes to:
        i. *Node 1:* subscribe to an image topic, resize the image to a 300x300 (as in *Task2.b*) and publish it on an output topic.
        ii. *Node 2:* subscribe to the output topic of *Node 1* and use the OpenCV functionality from *Task2.c* to process the received images. The processed images should be published on a third ROS 2 topic.
- **Bash:**
    h. Create a bash script that checks if the camera is connected. If so, the script launches the previous executables/launch files, otherwise prints out a warning msg.
- **Docker:**
    i. Create a *Dockerfile* that can be used to containerize your computer vision application.

Task 4: Version Control

- **Git Version Control:** Commit your code changes for Tasks 2 to your Git repository using thoughtfully Git functionalities.

Task 5: Submission

a. Push your code to your repository.

b. Provide a link to your GitHub repository and a comprehensive README.md file explaining how to set up and run your computer vision application and how to build the Docker (only if Task2.h was completed).