a. install gitlab runner & register the same
        Project -> settings -> CICD -> Runners -> Specific Runners
        Ref: https://docs.gitlab.com/runner/install/linux-manually.html or
https://docs.gitlab.com/runner/install/linux-repository.html

b. protected branch Project -> settings -> Repository ->
c. maintainer access, add deploy key (ssh rsa public key from machine)
        Project -> settings -> Repository -> deploy key
        https://docs.gitlab.com/ee/ci/ssh_keys/index.html
        https://docs.gitlab.com/ee/ssh/index.html#generate-an-ssh-key-pair
        Add to EC2 ~/.ssh/authorized_keys (public key) and add the same to gitlab SSH keys in
profile settings / preferences to connect to gitlab.
        Add generated private_key to ssh_agent
            eval $(ssh-agent)
            ssh-add ~/.ssh/id_rsa
            ssh -T git@gitlab.com
d. sudo gitlab-runner register
        register: https://docs.gitlab.com/runner/register/index.html
            repo domain, token, tags- **add branch or tags**, runner name, executor - docker,
        docker image ruby:2.6
f. sudo gitlab-runner start
g. check the new runner with green clr sts in repo settings ..
i. create CICD config file & add relevant cmds *
h. add env variables by referring below **
j. Install docker - https://docs.docker.com/engine/install/ubuntu/ (to be used by runner to execute
the pipeline) and disable shared runners for free gitlab account.

        * .gitlab-ci.yml
web - react build
1. SSH connection
2. dir (repo) navigation
3. git stash, checkout & pull
4. npm i
5. build script

Server - node pm2
1- 4 above
5. pm2 restart * / pm2 eco file

** env variables - CI_BRANCH, CI=false in start or build script (frontend), DEPLOY_SERVER -
ip or domain, REPO_DIR_PATH - absolute path, SSH_PRIVATE_KEY (ssh rsa key which is
added to ec2 ssh/ authorized_keys)

executor should be made available manually:　　　n=$(which node);n=${n%/bin/node}; chmod -R 755 $n/bin/*; sudo cp -r $n/{bin,lib,share} /usr/local;