Steps to rebase

1. List down the files which are being modified and what changes we are going to push.
2. Check the coverage (unit test) for each file and verify the target build. Take backup of the folder (if everything is proper).
3. Check the commit ids in git bash and git lab. The commits should be done on top of latest merge. (List down the ids for better understanding).
4. Carefully observe the changes being made in the files(which are to be pushed).
5. Check the author, through whose ids we are going to commit.
6. Go to the branch which is to be rebased.
7. Commit and push all the modifications.
8. Rebasing from branchA -> branchB

git rebase branchB branchA

1. git log and check if commits and rebase is successful(check the commits, it should point to latest merge).

8a- If merge conflict arises, go to the codebase and open GitExt Open repository application

8b- First select the branch to be rebased(branchA), then select the upstream branch(branchB).

8c- After selecting both the branches, right click and go to rebase current branch on -> selected commit

1. Check all the commits. All the commits should be observed carefully.
2. Push all the commits to respective master branches.
3. Go to git lab and check whether rebase is done properly.

* Go to graph. Your branch should be on top of the master branch(the branch onto which your current branch is rebased).

1. Check all the commits and ids.
2. Check the coverage and it should be same as before rebasing.

Issues might be faced:

1. SSH Key expiry- Generate a new ssh key in git lab.
2. While pushing the changes, we might encounter an error(unable to pus few refs)

* Check the previous commits. All commits should be in proper format.([Ticket id ] Message)
* If everything is correct, use git push -f command.