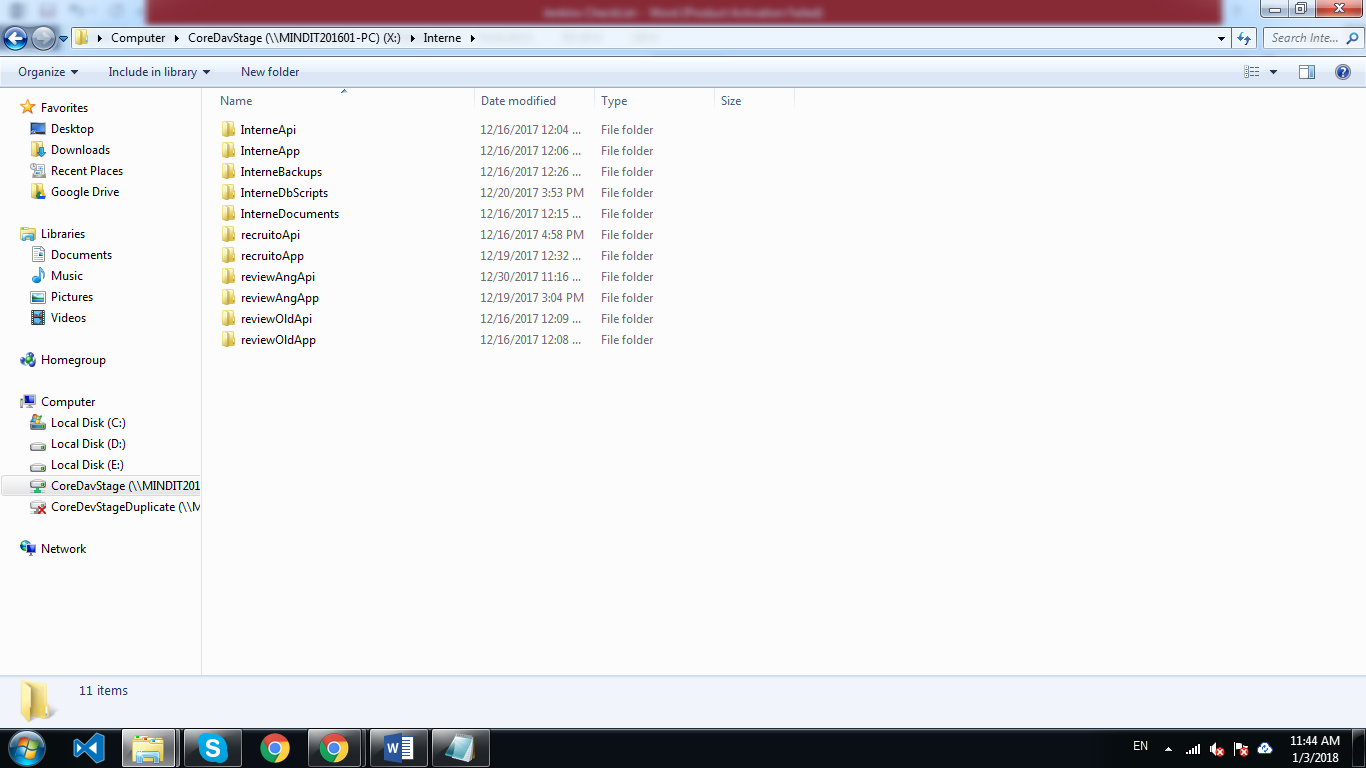
**.Net and Angular Projects:-**

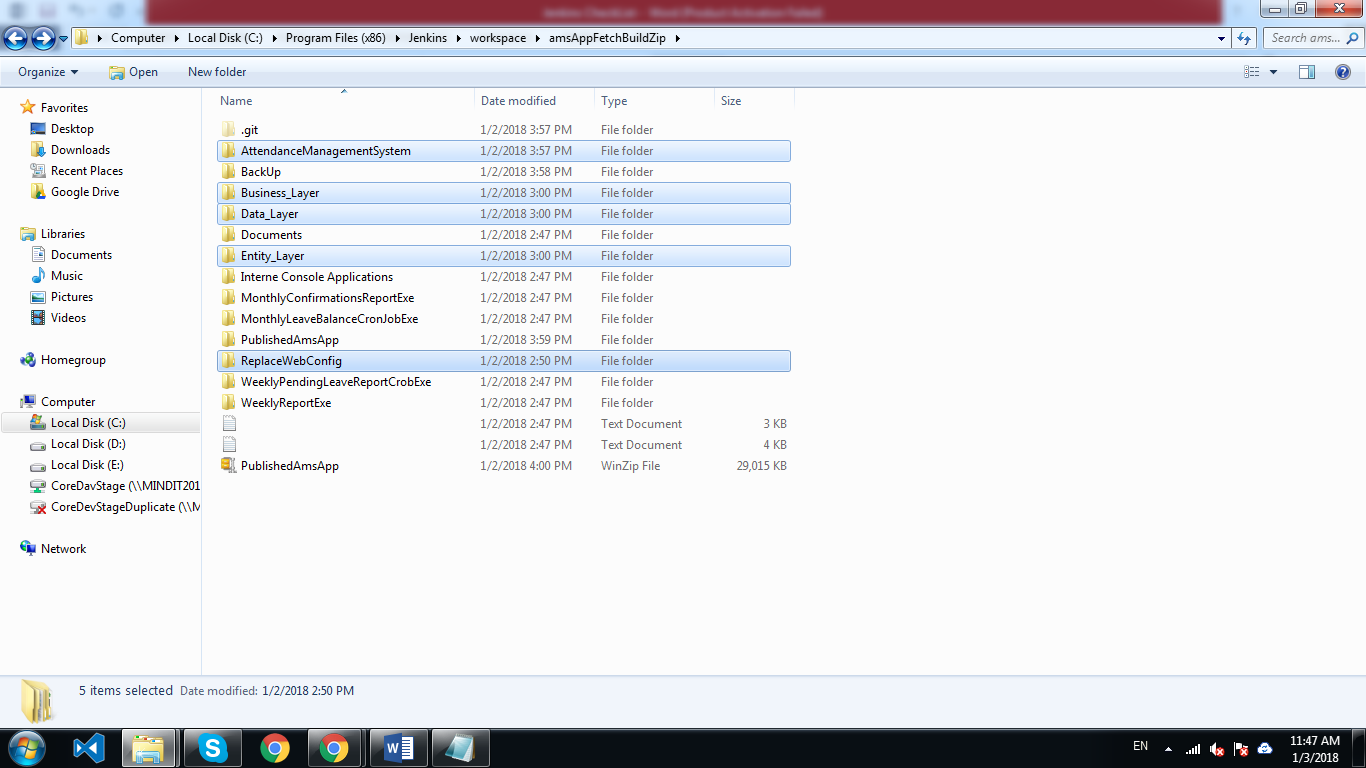
* Git structure should be
  + Main folder
    - App folder
    - Api folder
    - Db folder
      * Current.sql
    - Backup folder
      * Your backups
    - Documents folder
      * Your documents
* Supply web config for respective environment in respective branch(no need for push ReplaceWebConfig folder).
* Inside db folder a file name as Current.sql should be there and have script which needs to execute push it in respective branch.(first you make changes in current.sql,then push it in git, then after the release change the name of Current.sql as per your tag and make a new empty Current.sql where you can enter new changes of db)
* Commit git with tags.
* Provide deployed folder name if you made it on servers.(please don’t save backup on servers)
* If your project is of layered architecture then made the output path of all different layers to bin folder of main project in release configuration and if it is not layered then leave it as it is.
* Supply dependency dll’s in bin folder when push in git respective branch.
* ReplaceWebConfig folder must be present at parent location in app or api folder.
* Include all hardcoded/extra folders/files into your project/solution.
* If make any changes on your local webconfig, please make sure to do same changes on replacewebconfig also.

**Some screenshot in order to understand folder structure explained above:-**

This is inside git parent folder is Interne

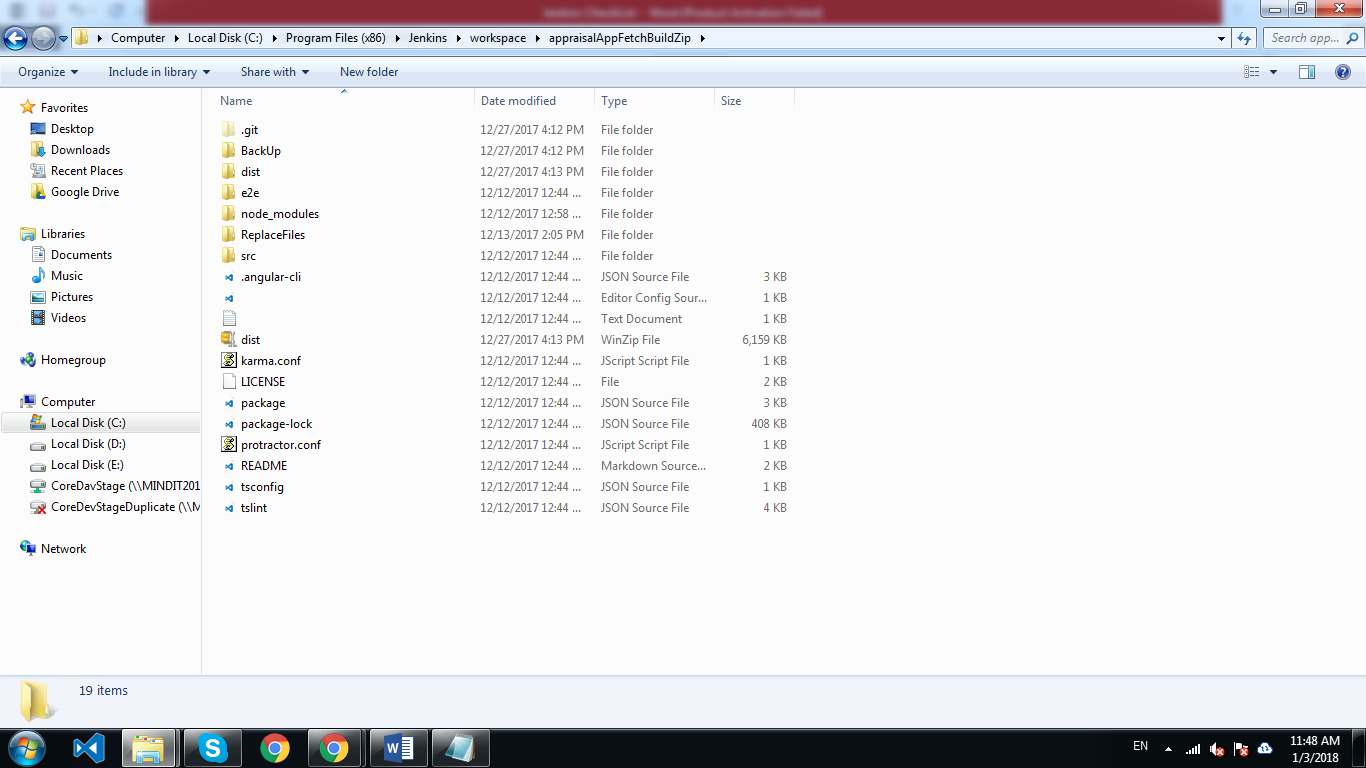


Layered .net project app after fetching from git should be pushed like this:-



Imp folder for you is highlighted

Angular project :-



* **Points to remember inside Jenkins workspace(not for developers)**:-
* For angular, npm must be installed at workspace,it will install node modules there which in turn enables to build the application using ng serve and if you made any changes to node modules then you need to make those changes to node modules inside Jenkins workspace manually.
* For .net projects we build app using MS build which is already setup in Jenkins.
  + In layered architechture we need to build soln and publish .projpublish file
  + In non layered architechture we can build an publish both .sln file.