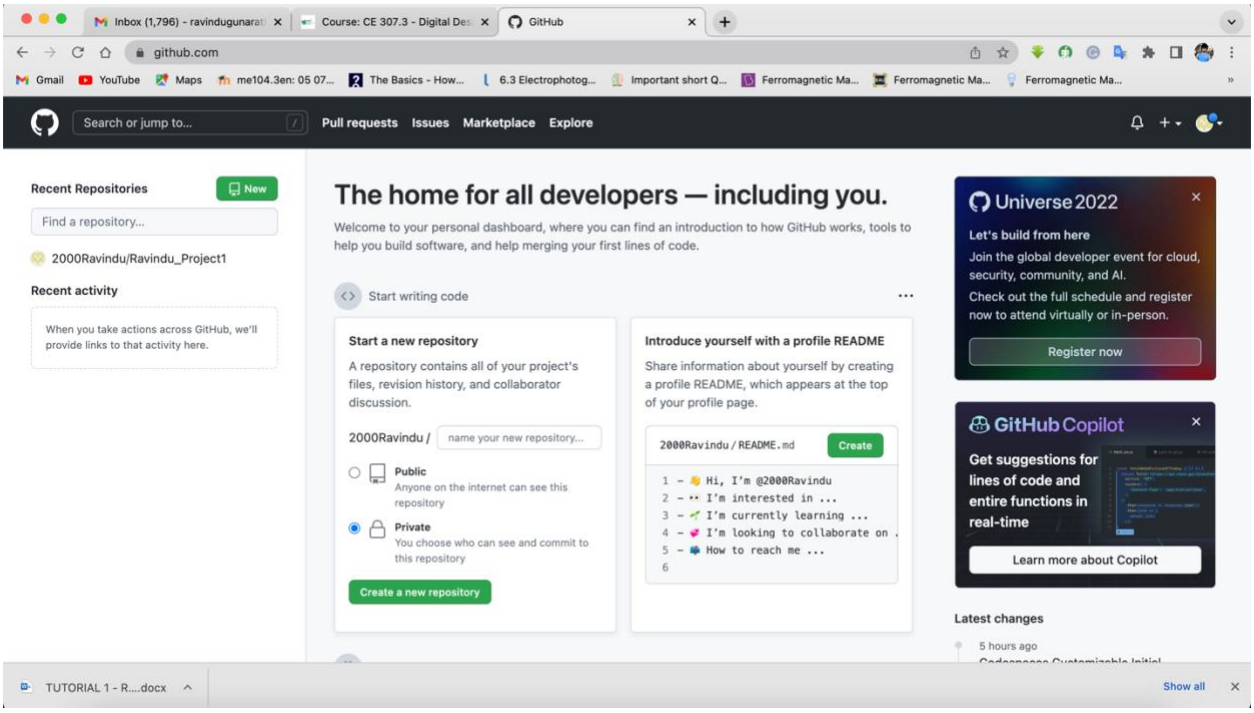
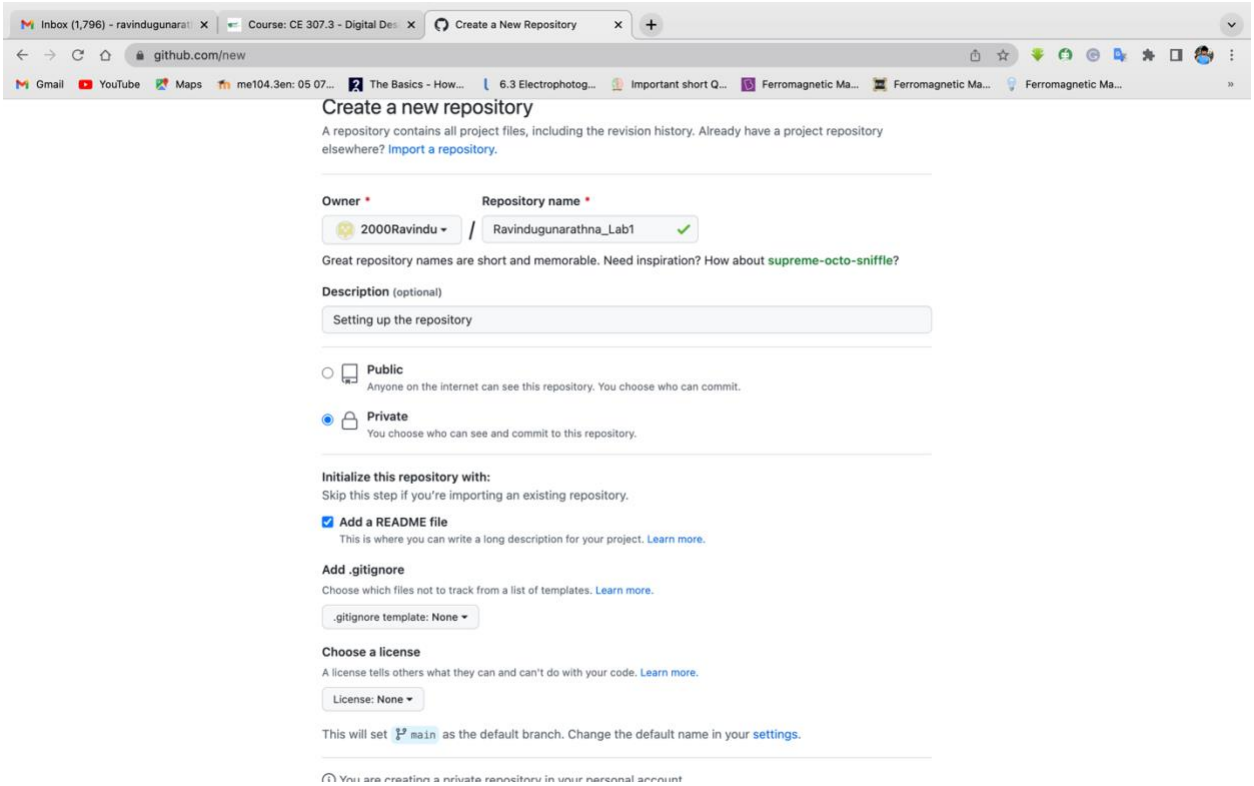
TUTORIAL 1: SETTING UP THE REPOSITORY

1.1 FAMILIARIZING WITH GITHUB

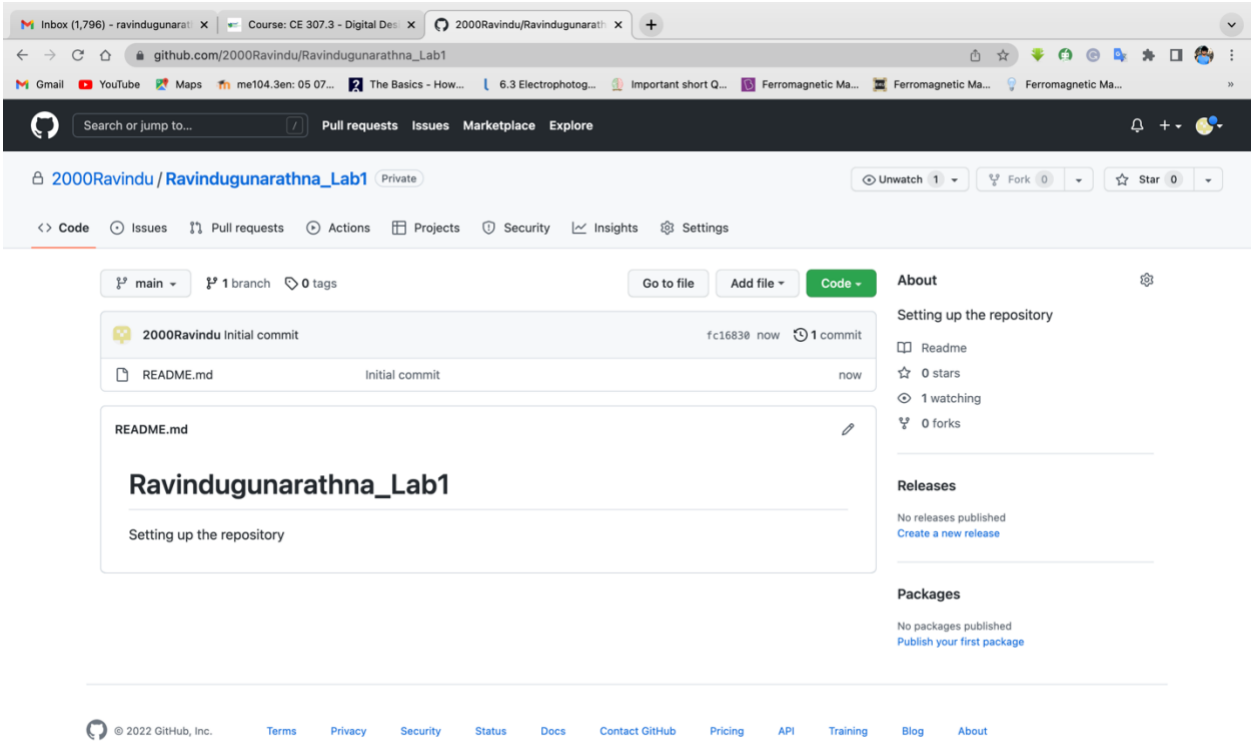
- Step 1 : Create your github account
  - Visit www.github.com
  - Create your account with a good username as you may need to use it in your future work



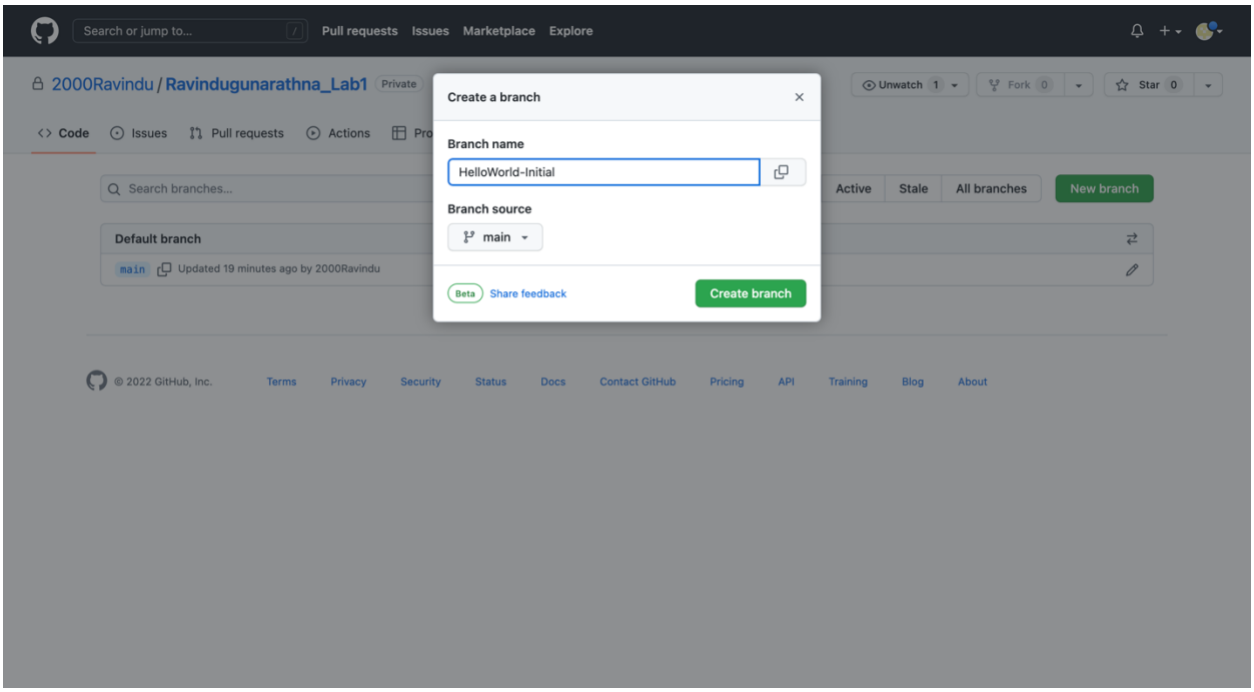
- Step 2 : Create a repository
  - A repository is used to host a single project which can contain any type of data including folders.
  - Login to your github.com page
  - By selecting '+' and New repository
  - Name of the repository "YourFirstName_HelloWorldLabs", Use a meaningful name with no spaces
  - Add a short description. Try to describe the project in short
  - Select "Initialize this repository with a README"



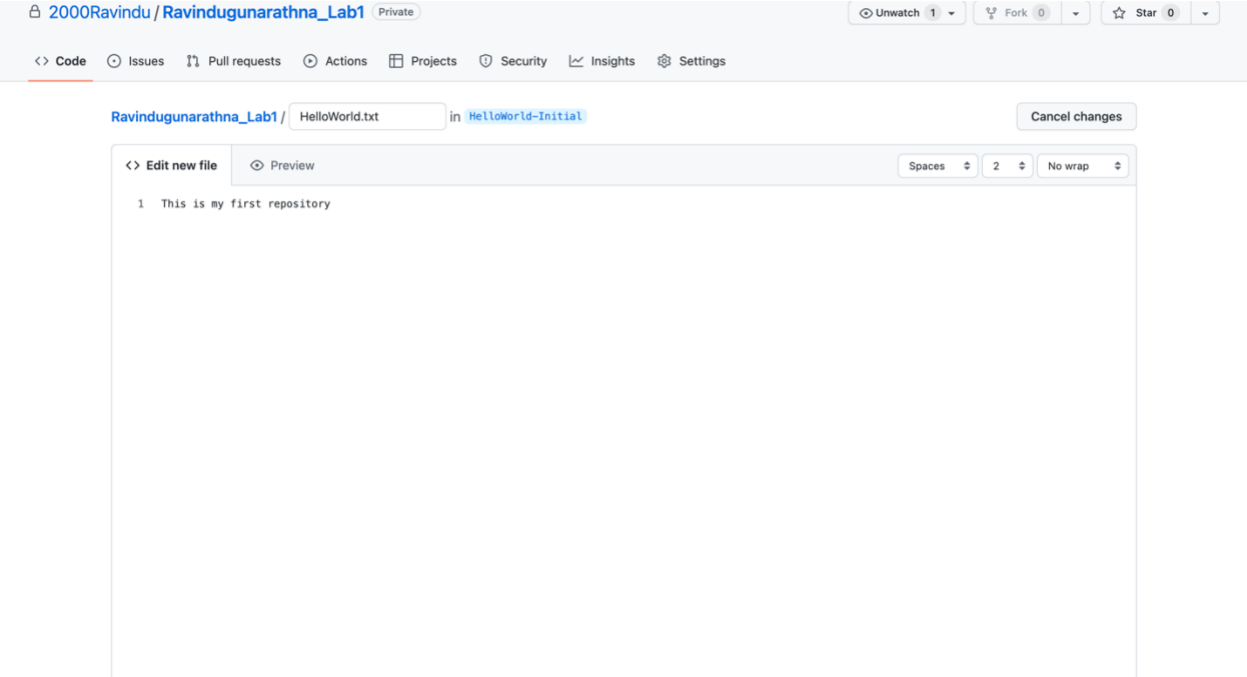
- Click create repository



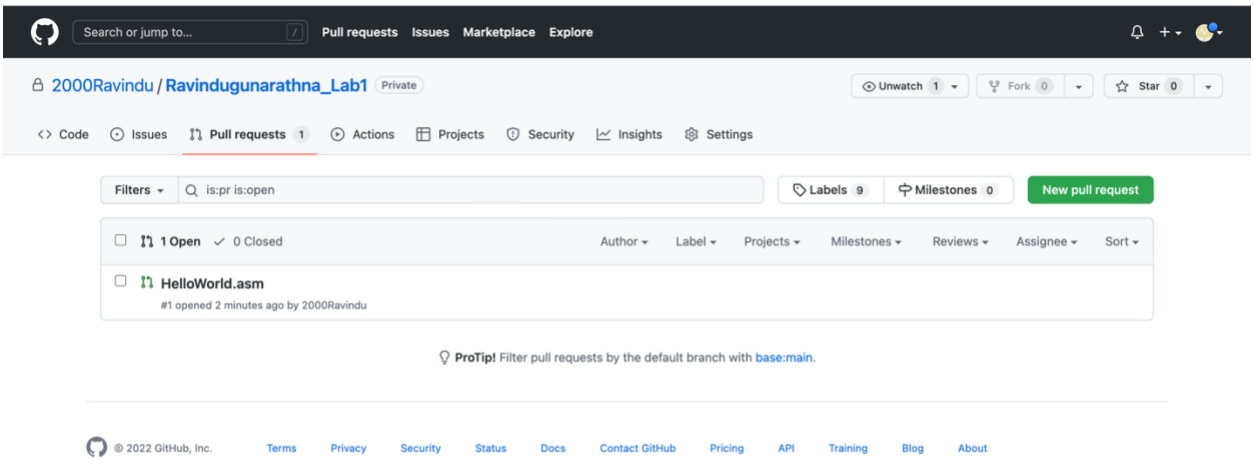
- Step 3: Branching
    - This is the way to work on different versions of a project.
    - The master branch is considered to be the working project which can be branched, edited and tested before committing back to the master branch. In this way master branch contains a working set of design files all the time. Your branch is independent from someone else's. Therefore it is possible to work simultaneously in developing different features of the same project.
    - Create a new branch named "HelloWorld-Initial"



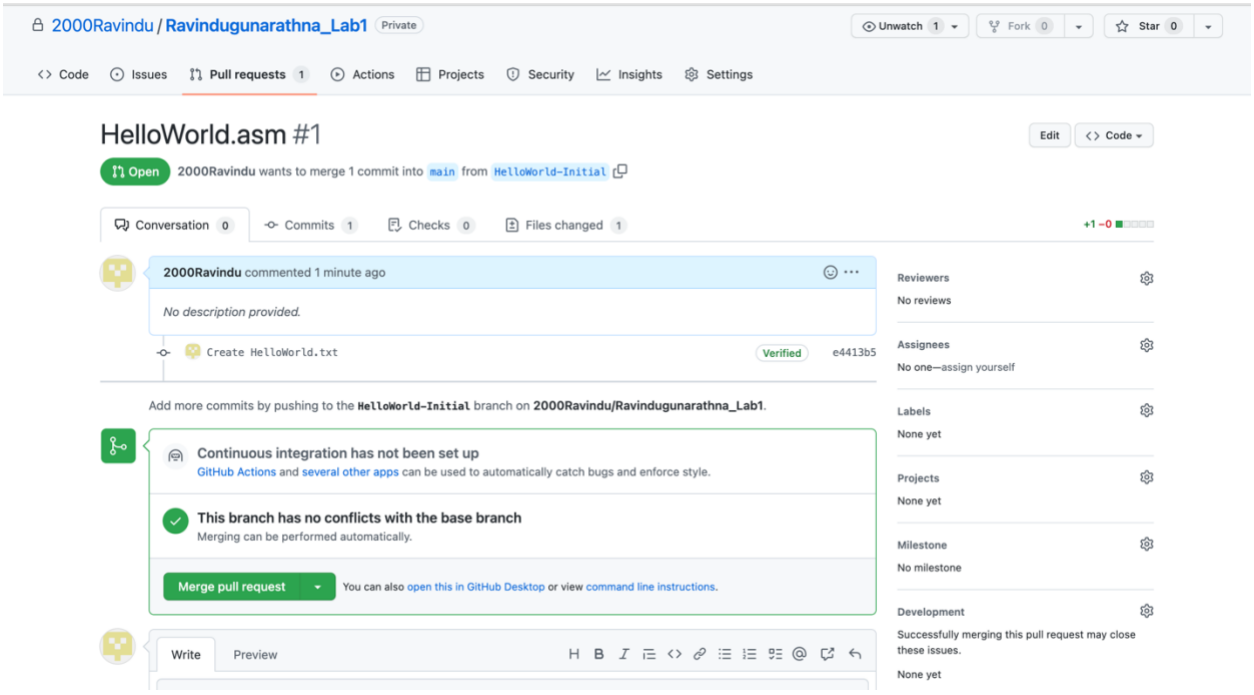
- Step 4: Commit changes
    - Now you can start editing the files in your branch
    - Add a new file named "HelloWorld.txt"
    - Write some text in the "HelloWorld.txt"
    - Then click "Commit changes"



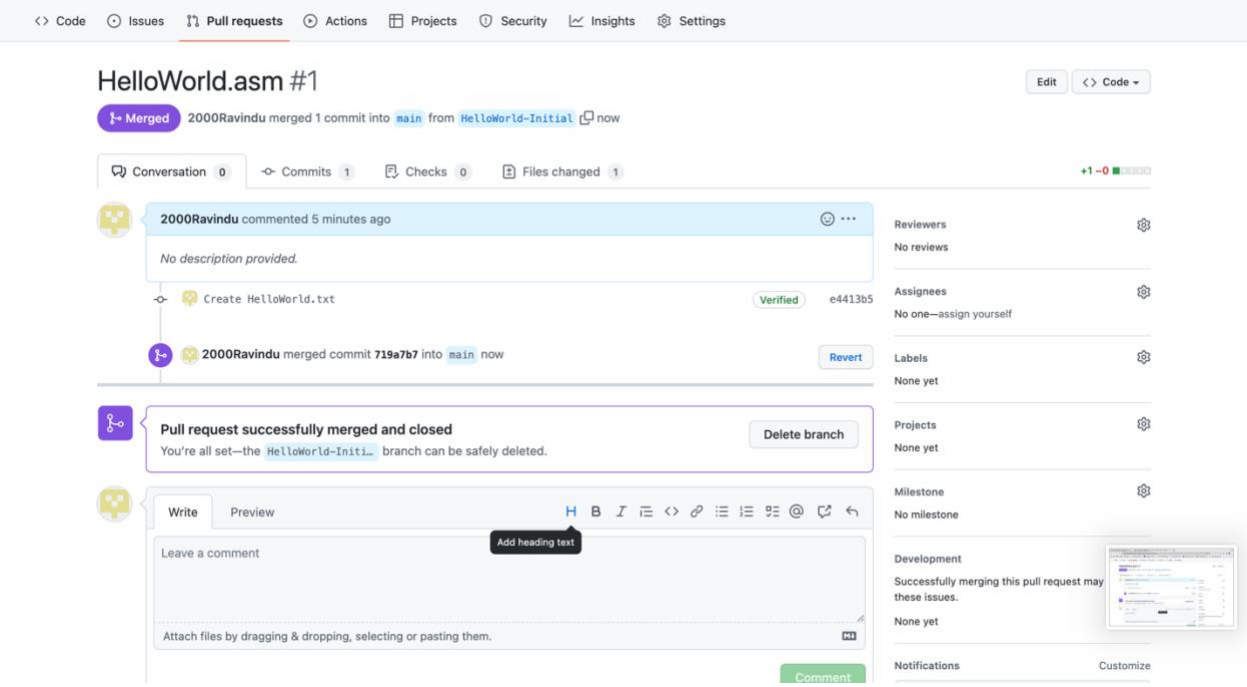
- Step 5: Open a pull request
    - Using pull requests, you can ask someone to review your contributions before merge them to the master branch or some other branch.
    - It is possible open a pull request as soon as you make a commit.
    - You can use “mention” system to ask someone to give feedback
    - It is possible to open up pull requests yourself and merge them yourself.
    - Create a pull request and give it the title “HelloWorld.asm created”



- Step 6: Merge the pull request
    - Once you finish testing your branch it can be merged to the master branch
    - It shows if there are any conflicts of the edited files. If two users edit the same file from two locations, there can be conflicts.



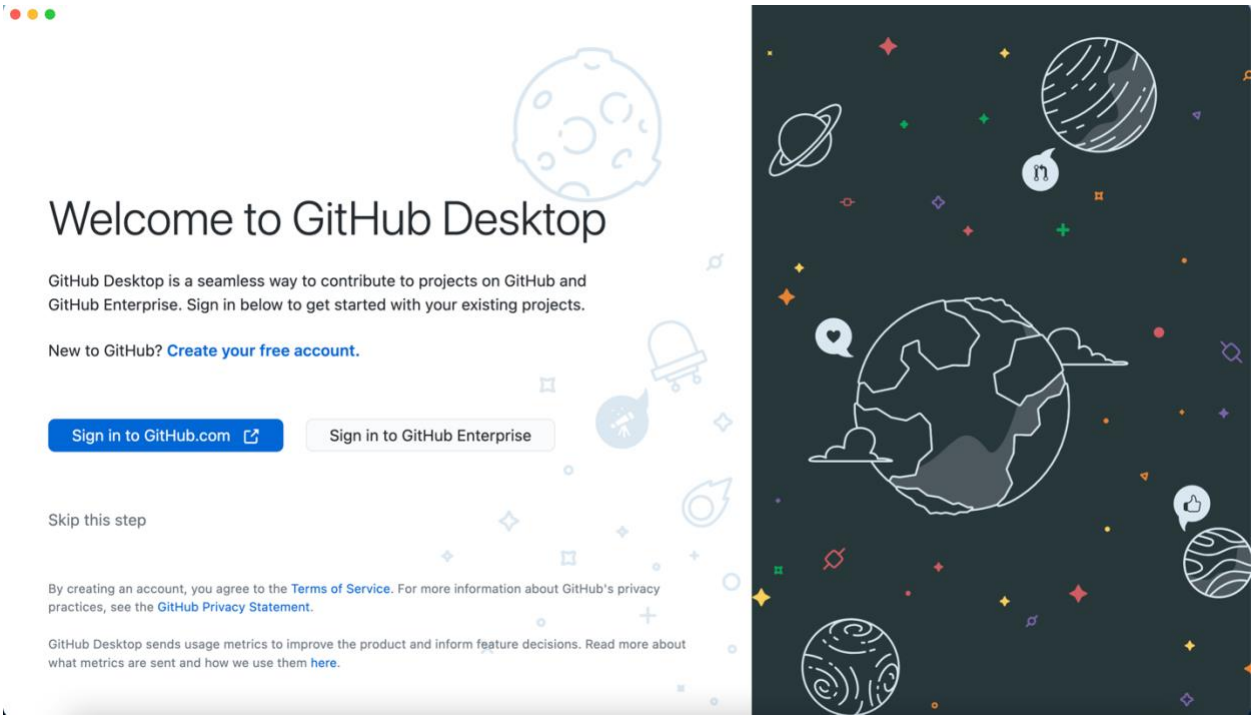
    - Click Merge pull request and confirm the merge



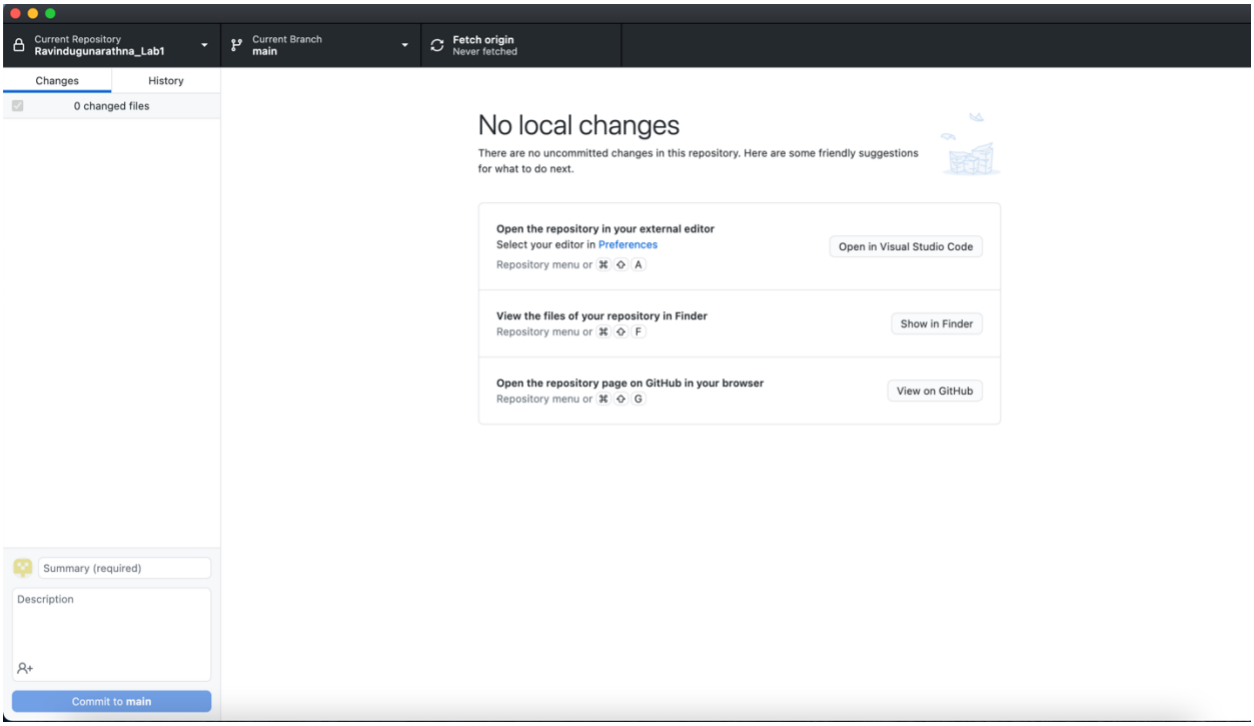
1.2 GITHUB GUI

You can manage your codes locally with the help of command line arguments or with the Github GUI. Following instructions are to setup the Github GUI in your computer.

- Step 1 : Installing Github desktop
  - https://desktop.github.com/ download and install Github desktop from here
  - Login using your Github account login details



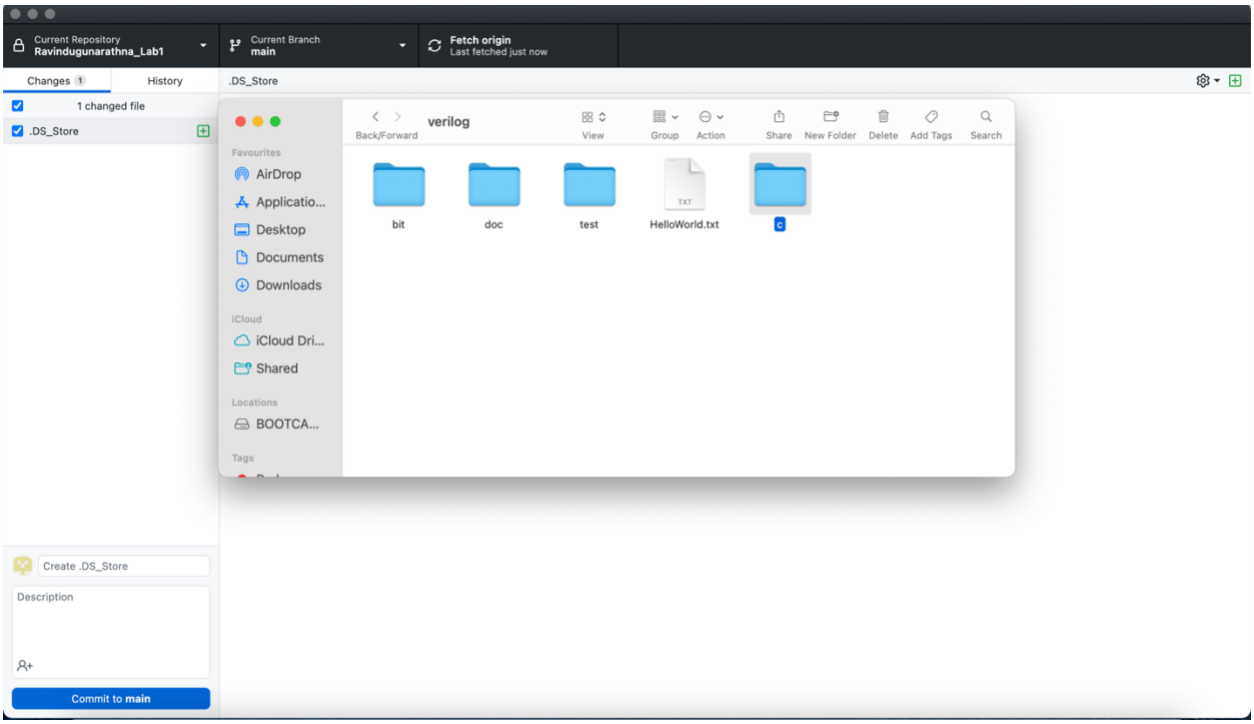
- Step 2 : Managing the repository
    - Clone the repository your created at Github.com to your computer



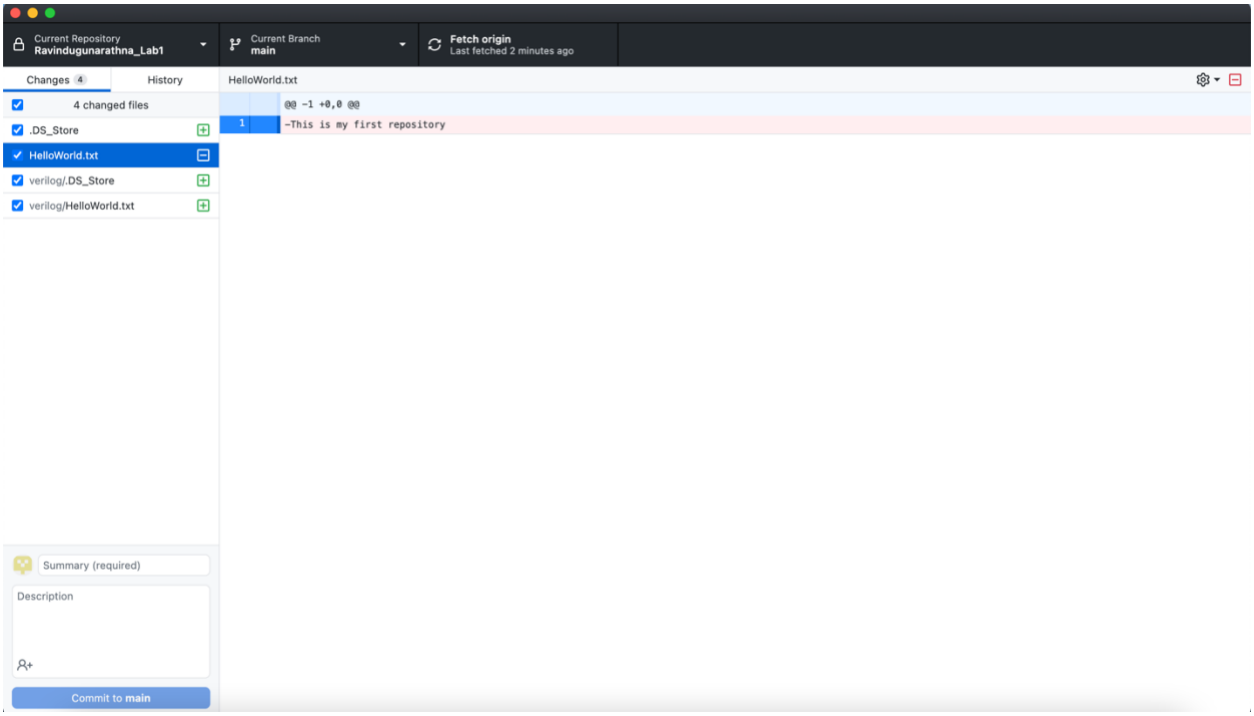
    - Now view the repository in file explorer
    - Create the following folder structure in your github folder
        - HelloWorld

- verilog
    - doc
    - test
    - bit

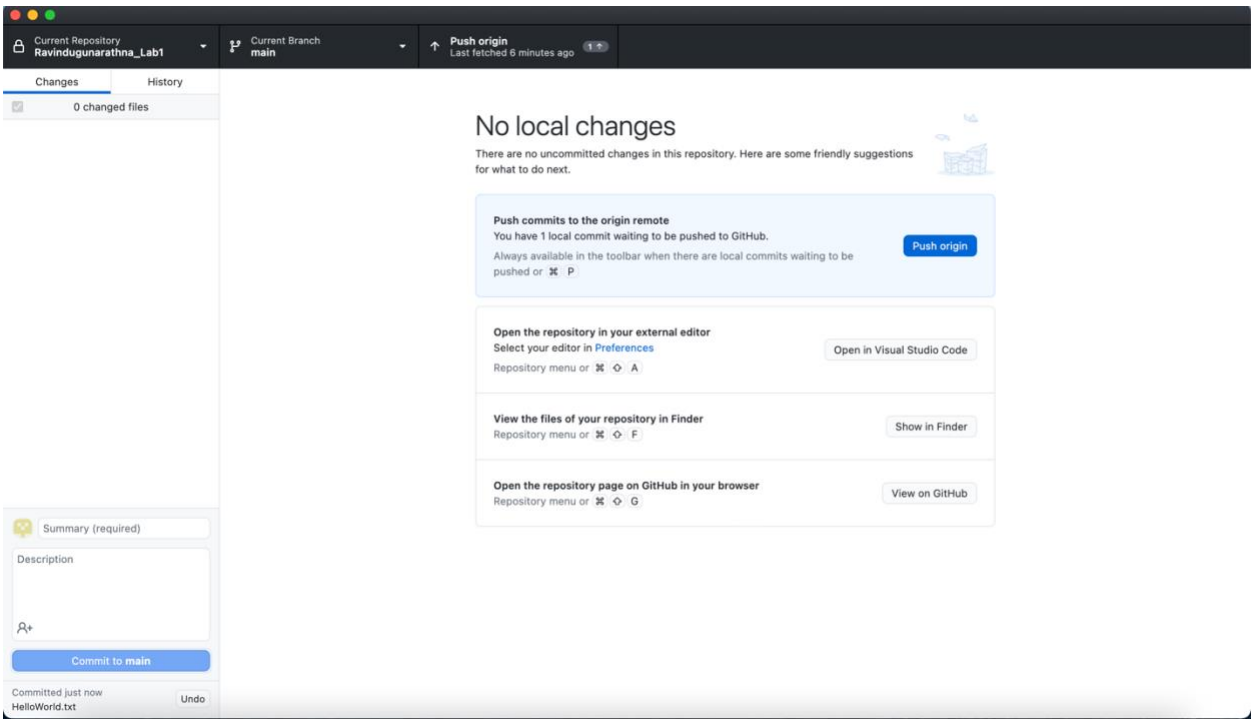
- Now drop the HelloWorld.txt file inside the c folder.



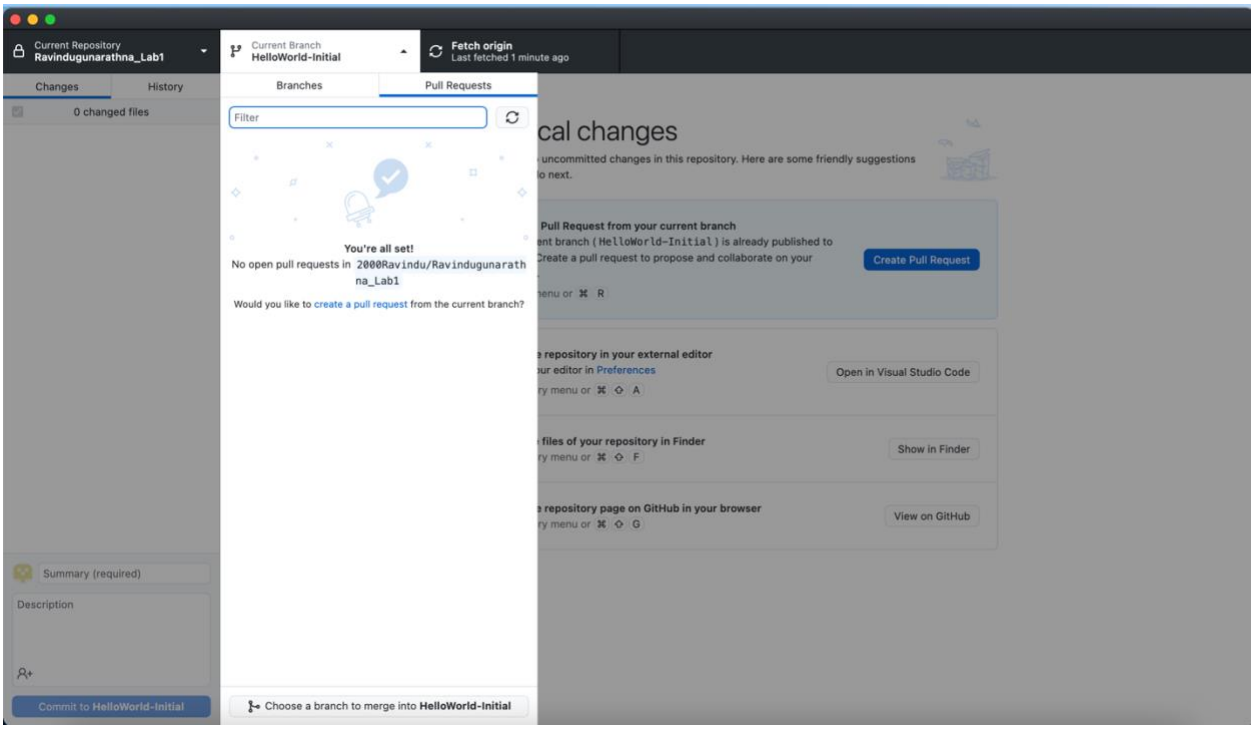
- Create some text files inside other folders as well
- Now on the GUI you can see these added files
- It shows that we have moved the HelloWorld.txt file



- Add a “Summary” – Moved HelloWorld.txt file to the C folder
- Now Commit to the Current branch



- Click push origin to upload these changes to your online repository.
- You can make a Pull request which will take you to the online repo from the Github desktop. Create your pull request as given in the first section.



1.3 COLLABORATION

Main advantage of Git repo is the possibility of working simultaneously on different parts of a project. Follow the steps to create your own repo to be shared between team members.

- Step 1: Creation of new repository
    - Create a public repo for your group project
    - Use a shortened name as the repository name
    - Create this repository without any files by the group leader

- Step 2: Collaboration
    - Now under the created repo goto Settings and select Manage access from the left sidebar
    - Now under Invite a collaborator, add your team members.

- Step 3: Project setup
    - Please make sure to create a branch from the master. NEVR MAKE CHANGES TO THE MASTER.
    - Distribute different possible modules among team members. Eg: 4:1 mux, 4-bit adder…etc.

- Create a branch with a title like “SubProjectName-‘yourfirstname’”, like that create different branch among team members

- Step 4:
  - Commit your changes and create pull requests. Make sure to give a short description of the change you have done in the commit message.
  - Now the team leader can accept the pull requests if these changes are ok.
  - Once accepted check the master branch and observe the differences.

1.4 TASKS

1. Add “ishara0925” as a collaborator to your CLEAN project repository. This repository is the one you are going to use for your project.
2. Add screen shots of the repository folder structure to your project presentation. Project presentation should be present inside the presentation folder in your project folder.
3. Add a short description of your project under the repository description.