**Project Website on GitHub**

1. **Source and References:**
   1. YouTube Video with steps on how to setup the site.

<https://www.youtube.com/watch?v=c02lc-bxrGU&ab_channel=TechZ>

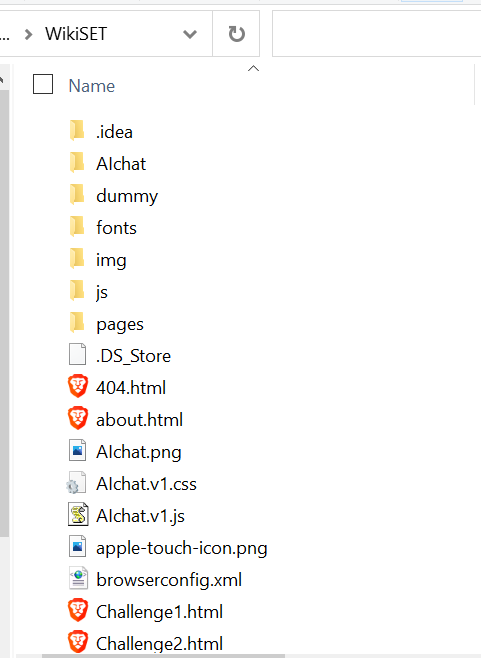
* Website with open-source templates for setting up webpages.

<https://html5up.net/>

* Template used for this project

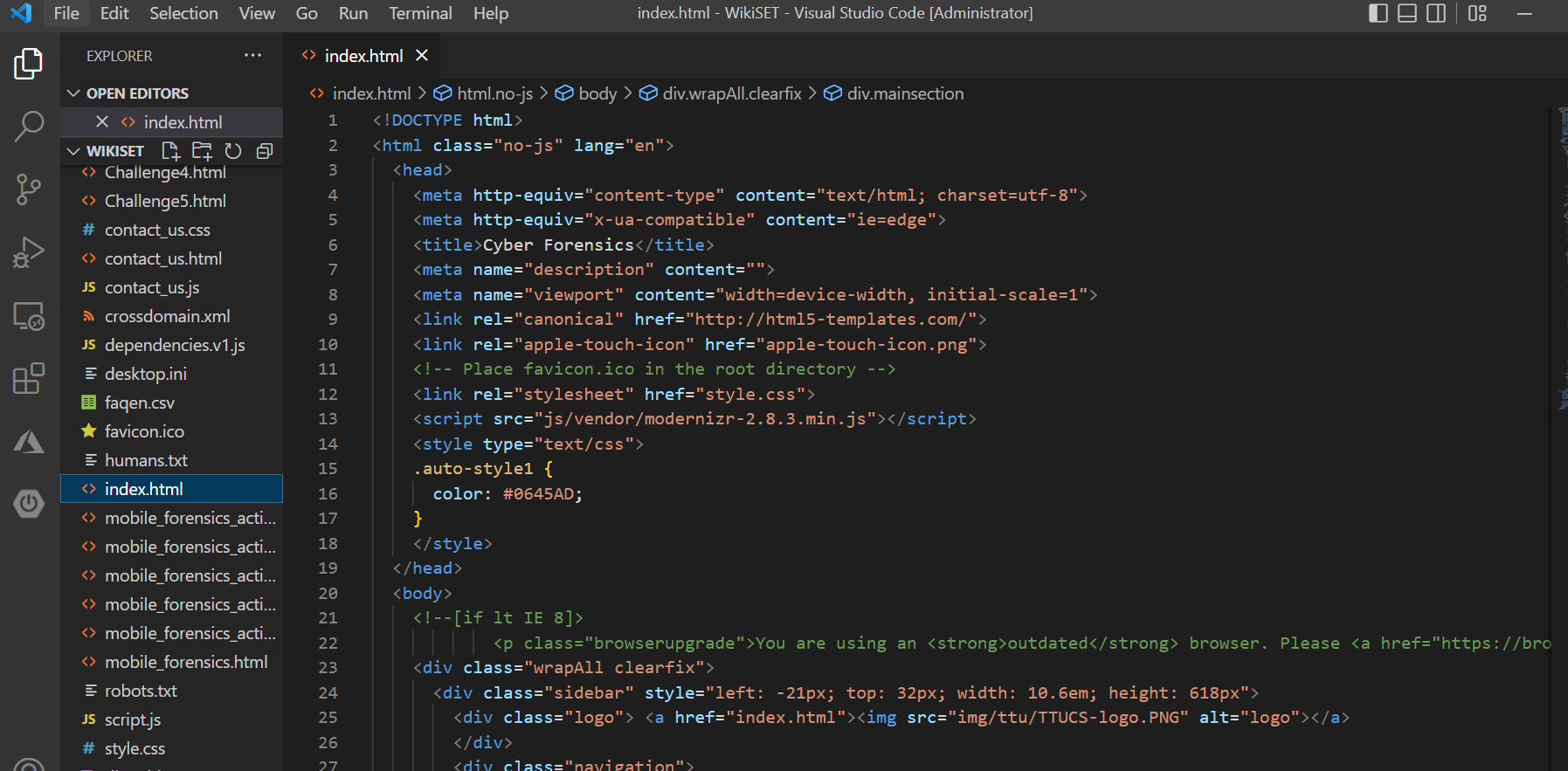
<https://github.com/asiamina/WikiSET>

1. **Steps on how the project is created.**
   1. Download the template zip file from <https://github.com/asiamina/WikiSET> on your local computer and open the files with any IDE. (I have used Visual Studio Code for editing the files).
   2. Extract the zip file to a folder and rename it (I have named the folder as “WikiSET”).

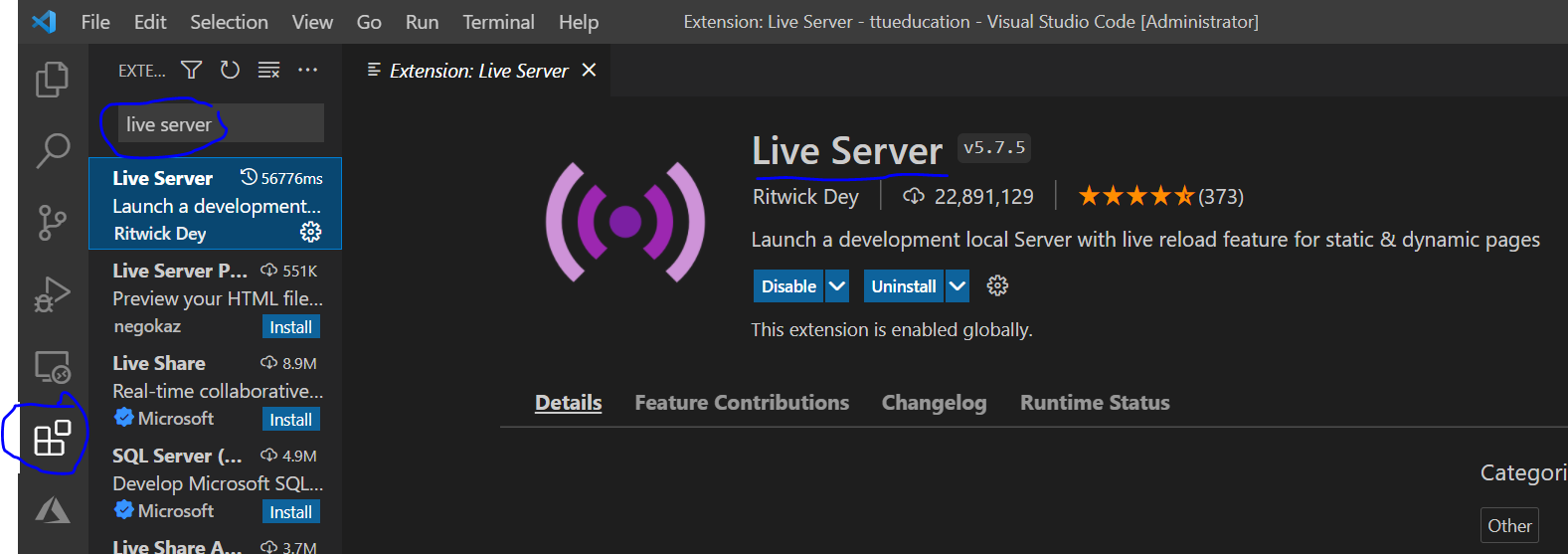


(Note: You can remove unrequired files like ReadMe.txt, License.txt from the folder)

* 1. Open the folder in IDE and check the files.

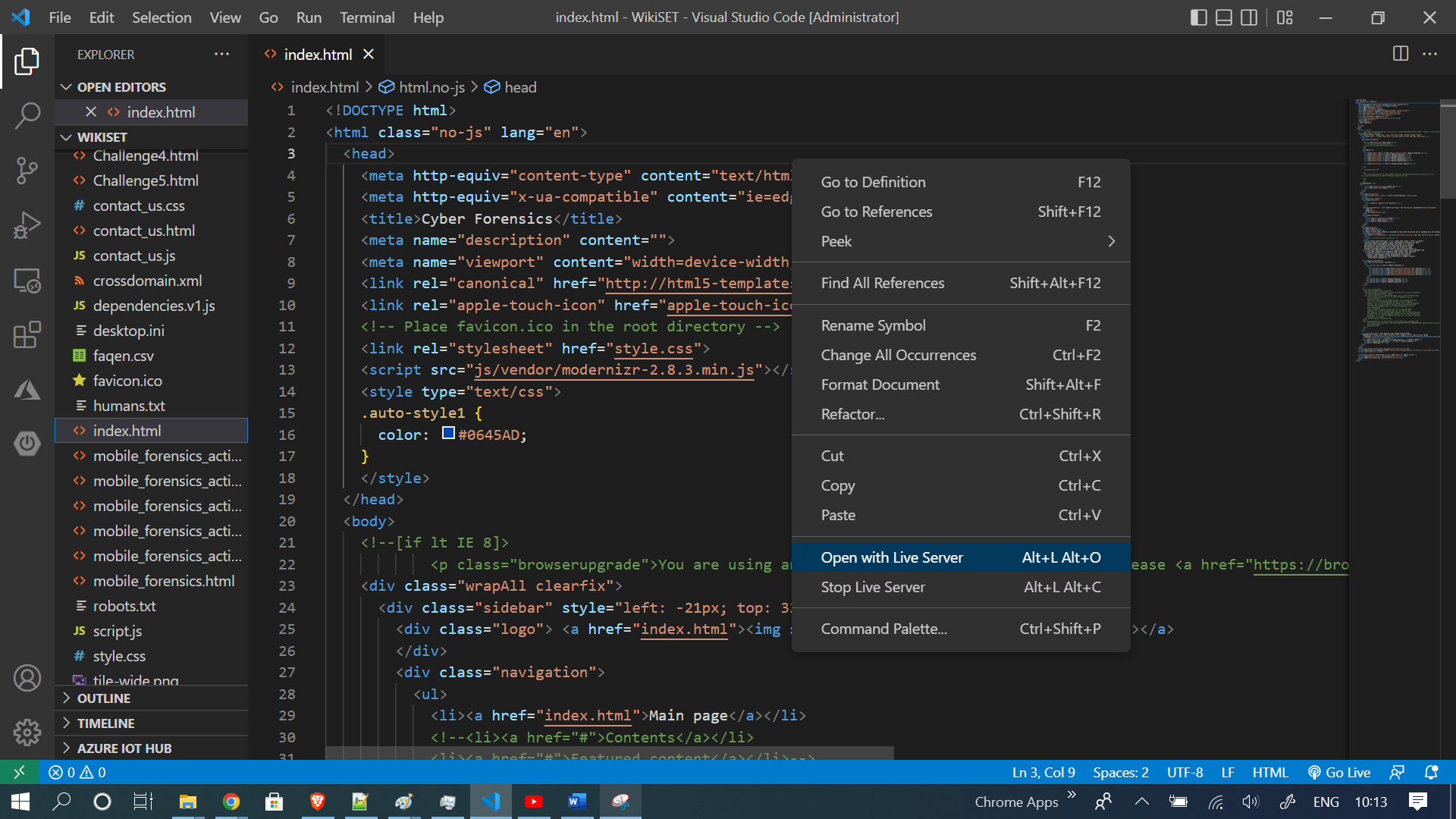


* 1. Go to Extensions and install “Live server” as shown below



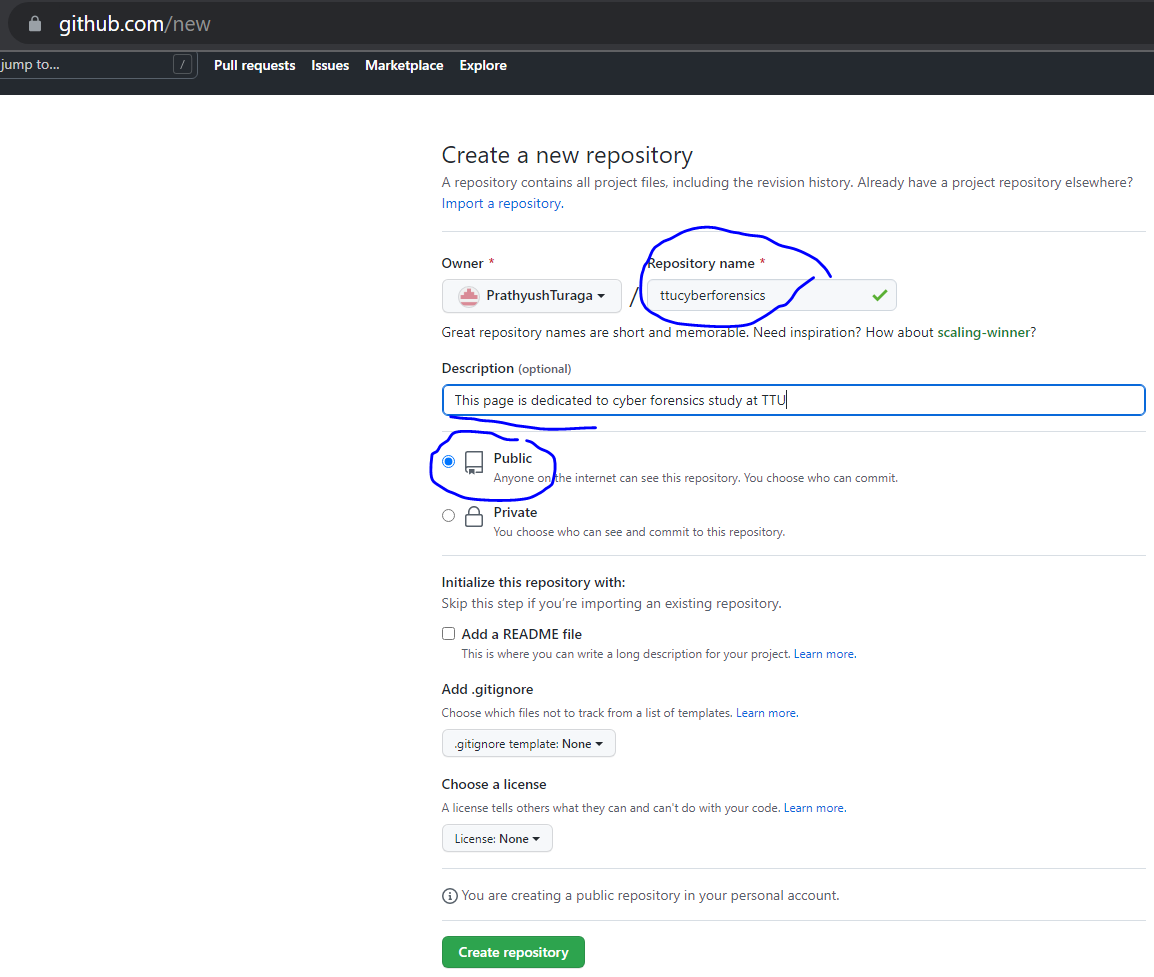
(Note: This screenshot belongs to Visual Studio Code)

* 1. After installing Live Server, you can come back to files location and open index.html file. Here you can run the file with Live server as shown in screenshot.

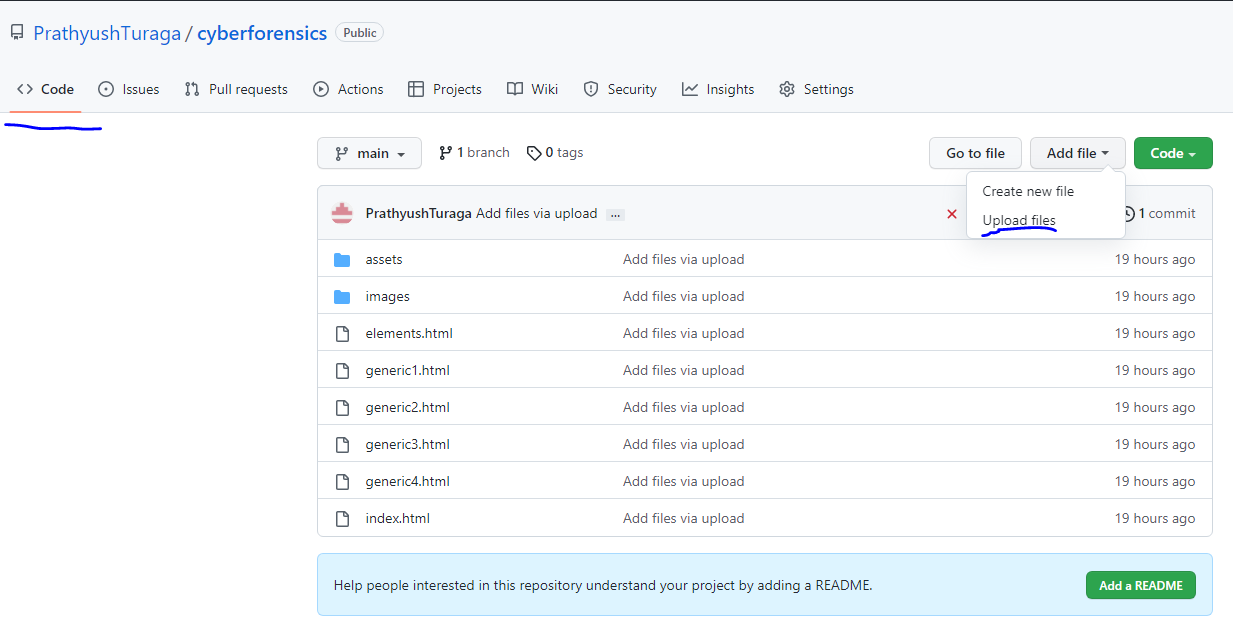


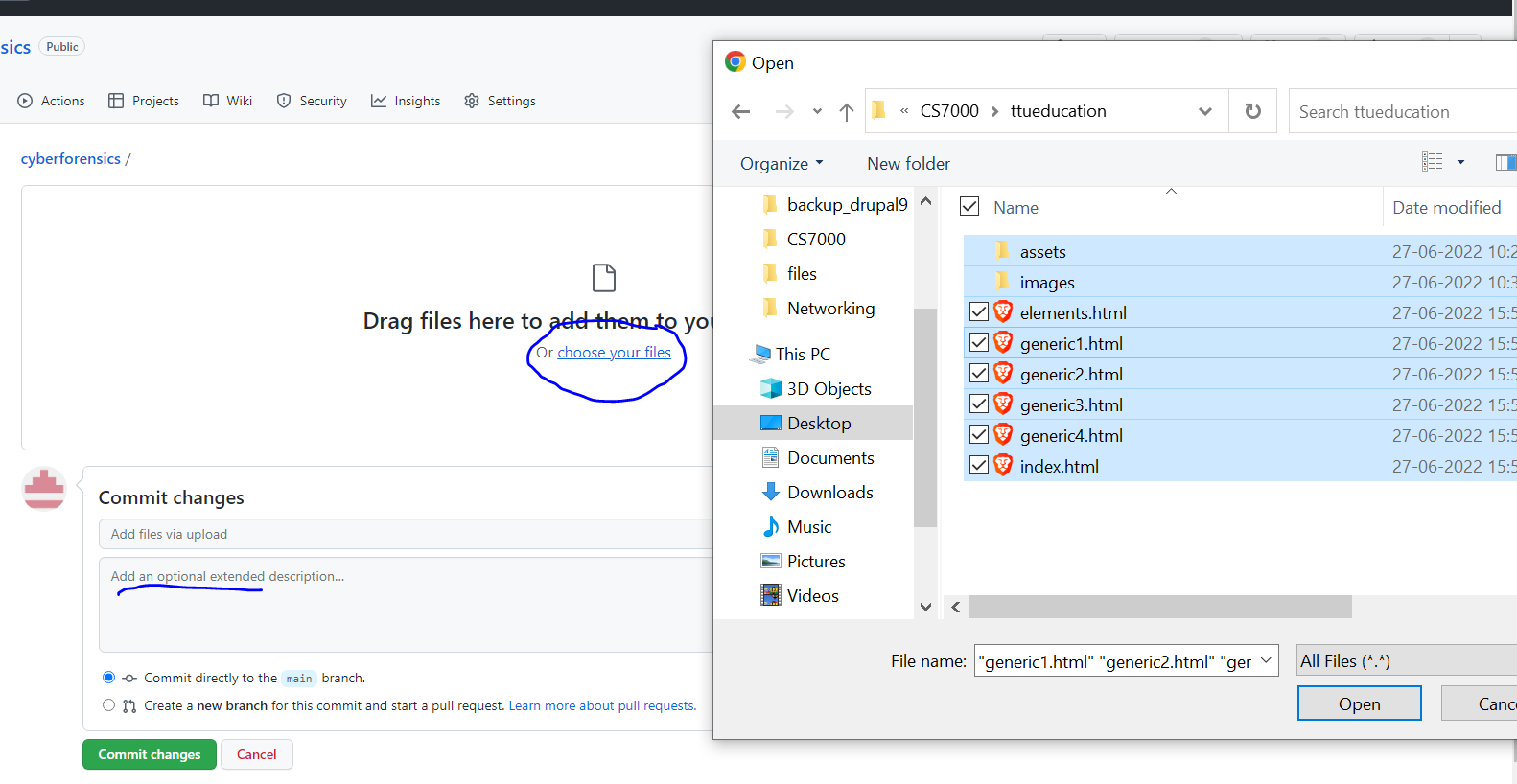
* 1. This will open a web server on port (5500 is default for Visual Studio Code) and you can access the default template website on <http://localhost:5500/index.html>
  2. Now you can start editing the HTML files and create required website.
  3. After all the changes are completed, you need to upload the files to GitHub with following steps.
  4. Login to your GitHub account and create a new repository.

(Put repository Name, Description, Public and leave all others to default settings)



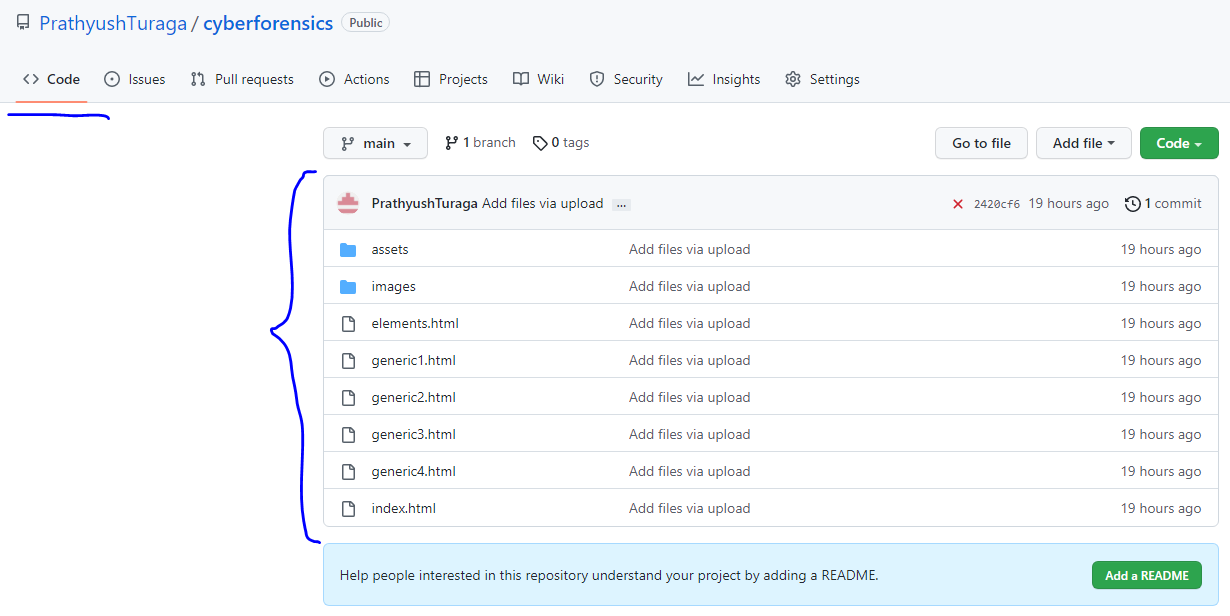
* 1. In the repository page, go to Code -> Upload Files -> (Upload all files from your local system)



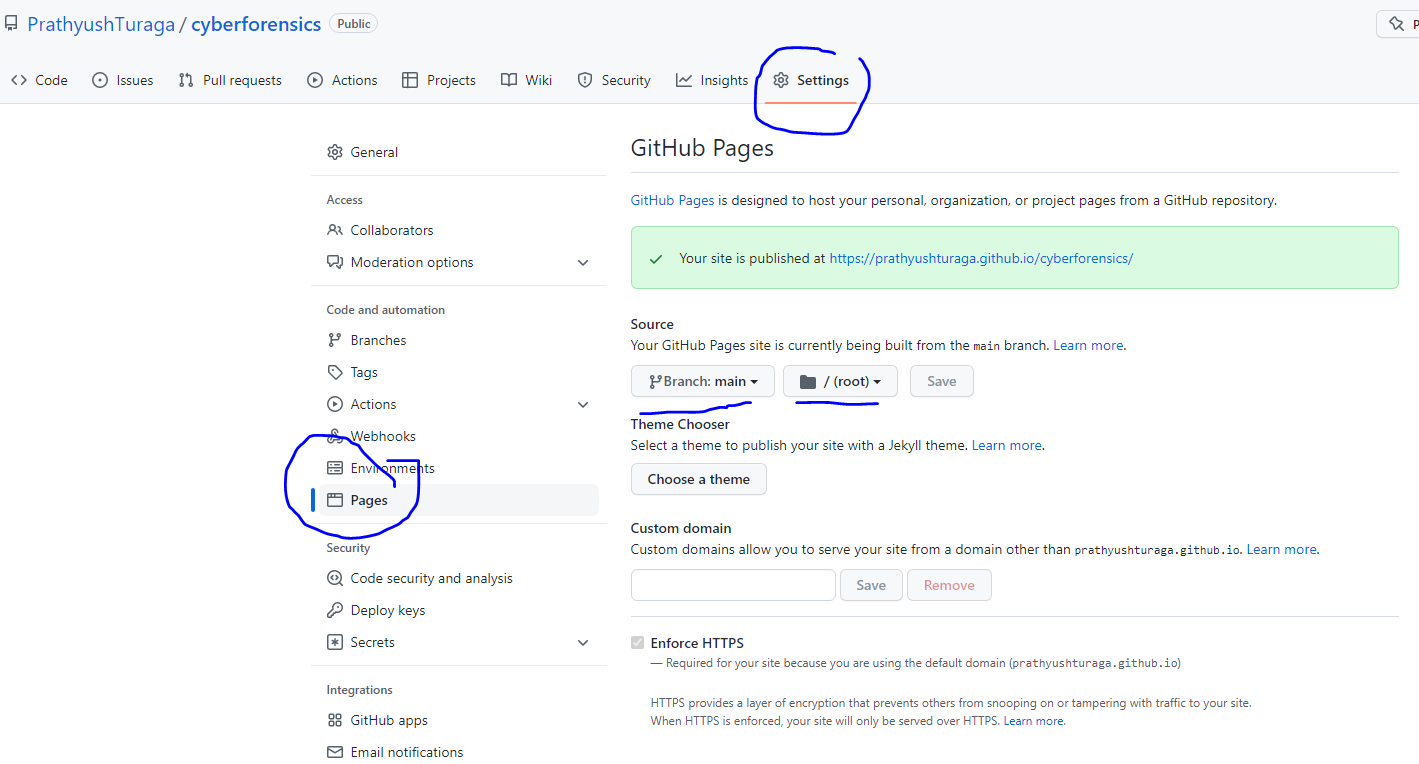


(Give optional extended description as “Initial Commit” for initial setup)

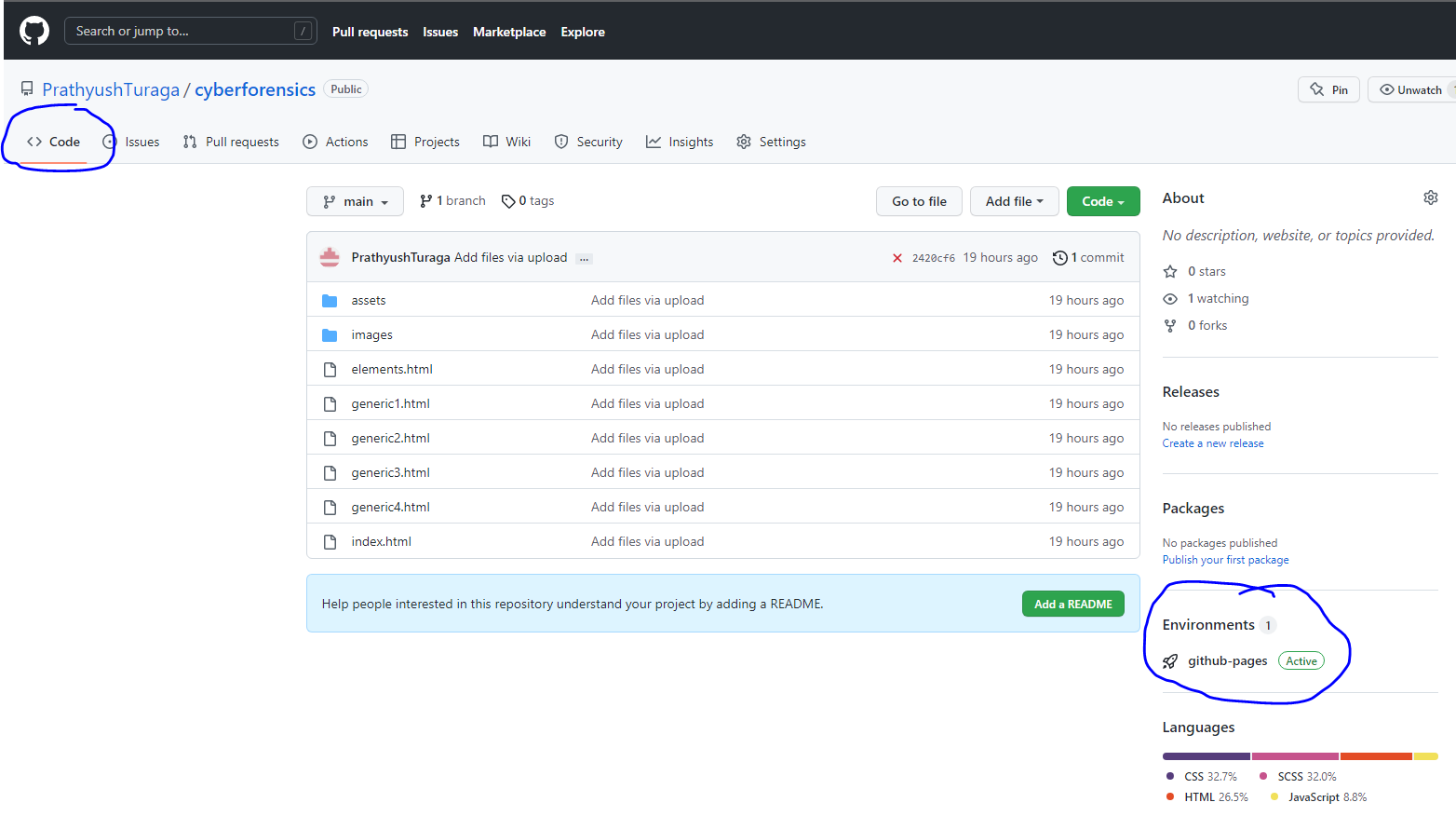
* 1. Now when you go to Code section of repository it will look like below.



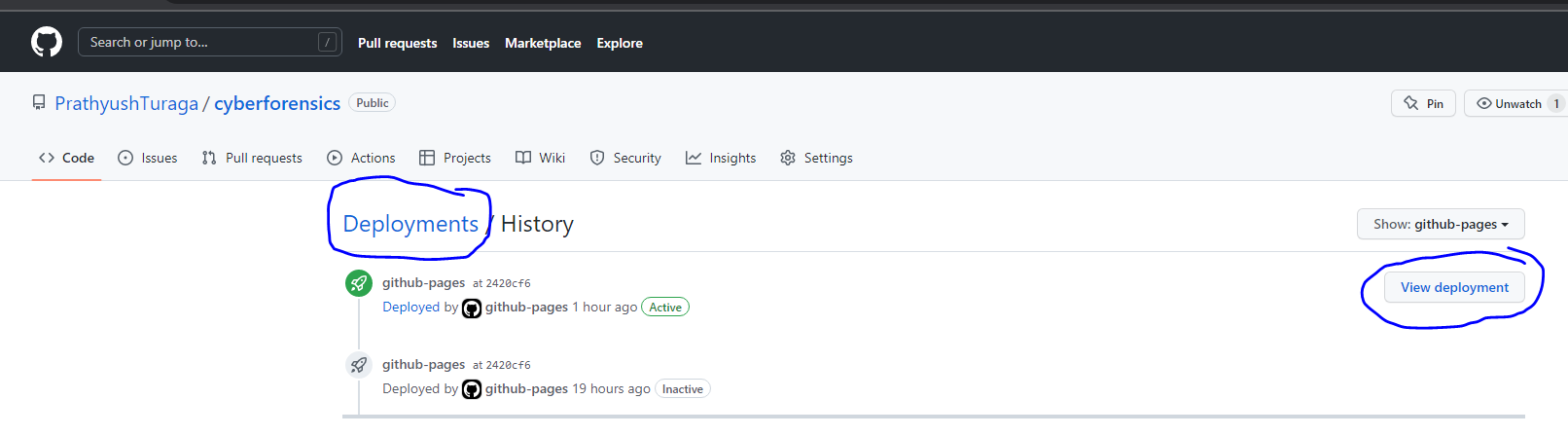
* 1. Now go to Settings -> Pages -> Source (Branch: Main, Directory: /root) -> save



* 1. Just above Source you can find the site URL for reference.
  2. Now go to Code section and click on github pages under environment as shown below.

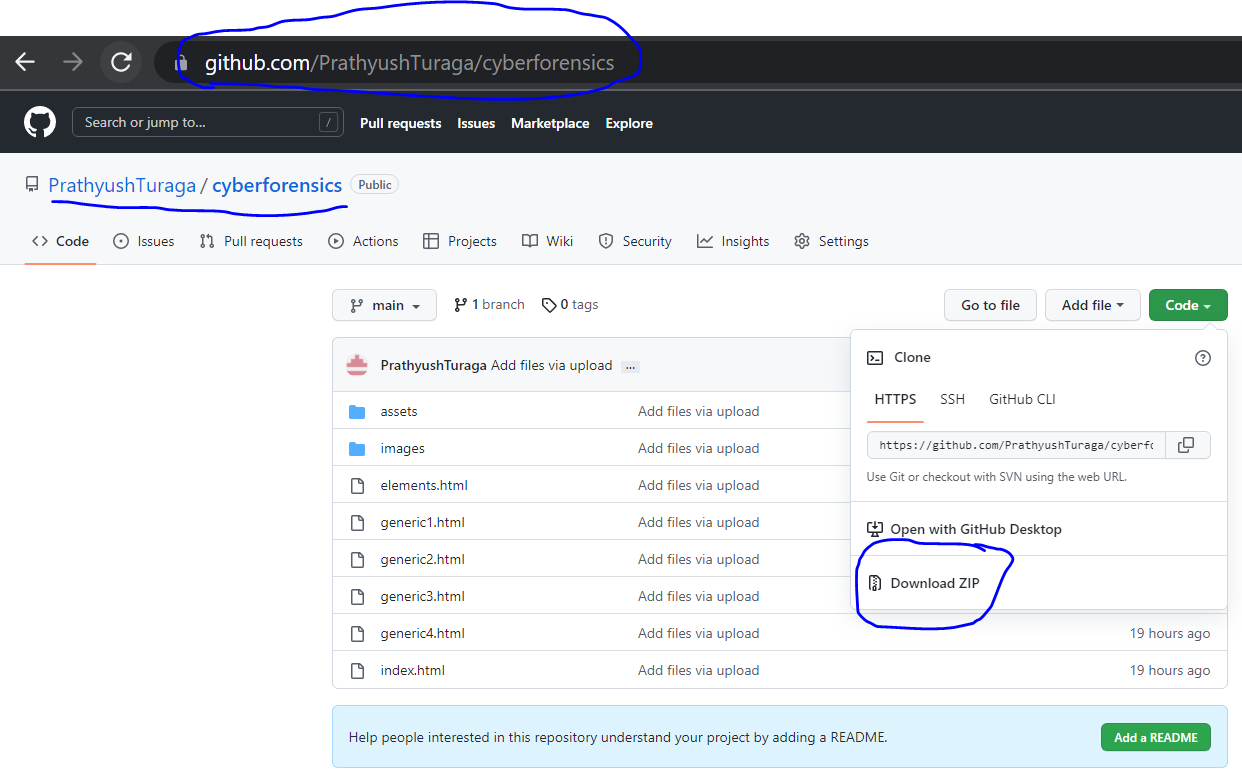


* 1. This will take you to deployments page as shown below



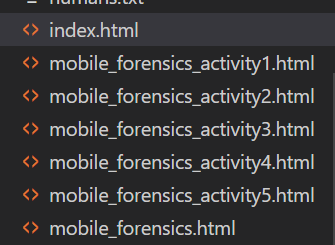
* 1. If deployment is successful, you can see the website up and running fine upon clicking “View deployment” button as highlighted in below screenshot.

1. **Future Development and addition of modules**
   1. Go to project URL on github and download the zip file to your local computer.

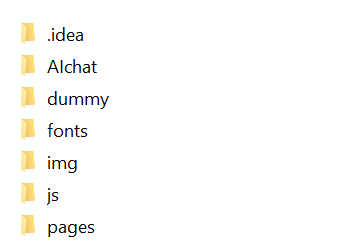


* 1. Extract the zip file to a folder and open the project in IDE (I have used Visual Studio Code). Please refer sections from 2.2 to 2.7 for setting up website on your local environment.
  2. In “index.html” file, there are sections referring to filenames “mobile\_forensics.html” which are topics on existing website.
  3. For adding new module, you can make a copy of the file “mobile\_forensics.html” and rename it as “network\_forensics.html” etc based on requirement.
  4. Inside “mobile\_forensics.html” file, customize your data sections with basic html knowledge.
  5. Now, inside “index.html” page, add new lines for “generic5.html” just beside other pages.

Example:



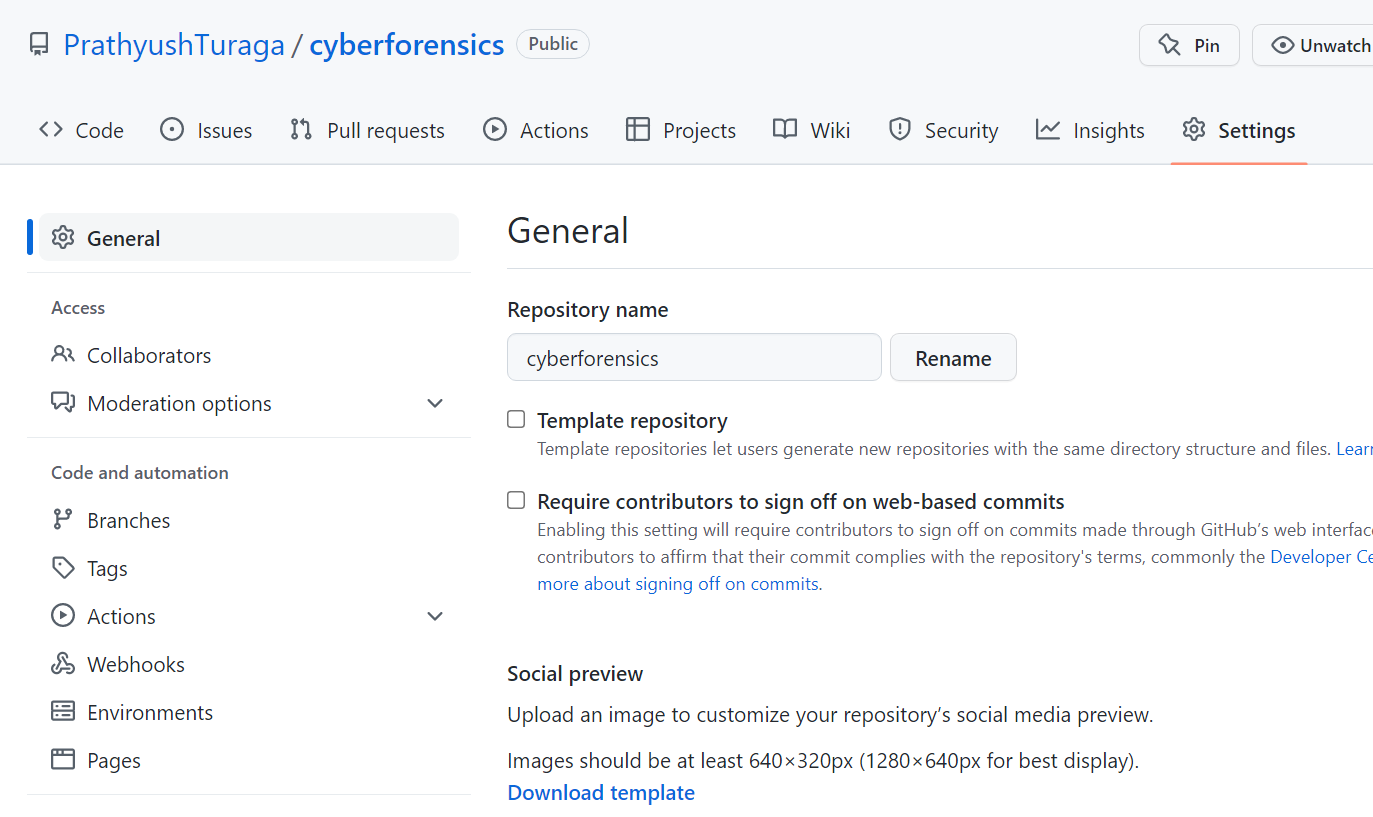
* 1. If there are any image specific files, add them to “img” sub-folder under the main folder and reference them accordingly in html code.

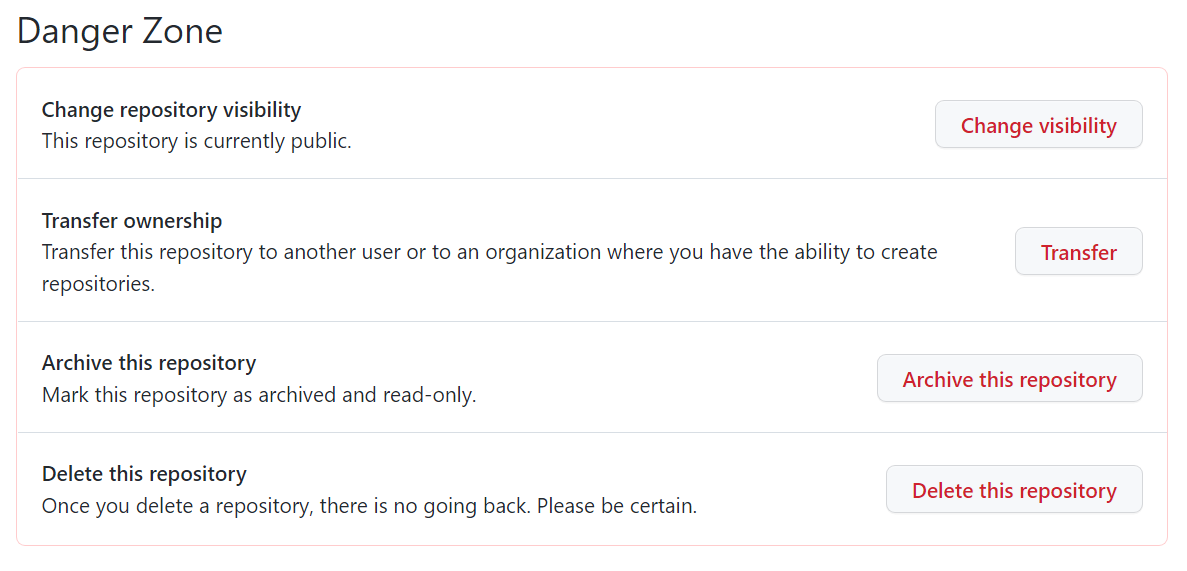


* 1. After all the changes are completed, you can check in the updated files to GitHub repository similar to sections 2.8, 2.10, 2.11.

1. **Security:**
   1. There are 2 options for repository accessibility i.e., Public (View access for everyone) and Private (Giving restricted view access to specific users).

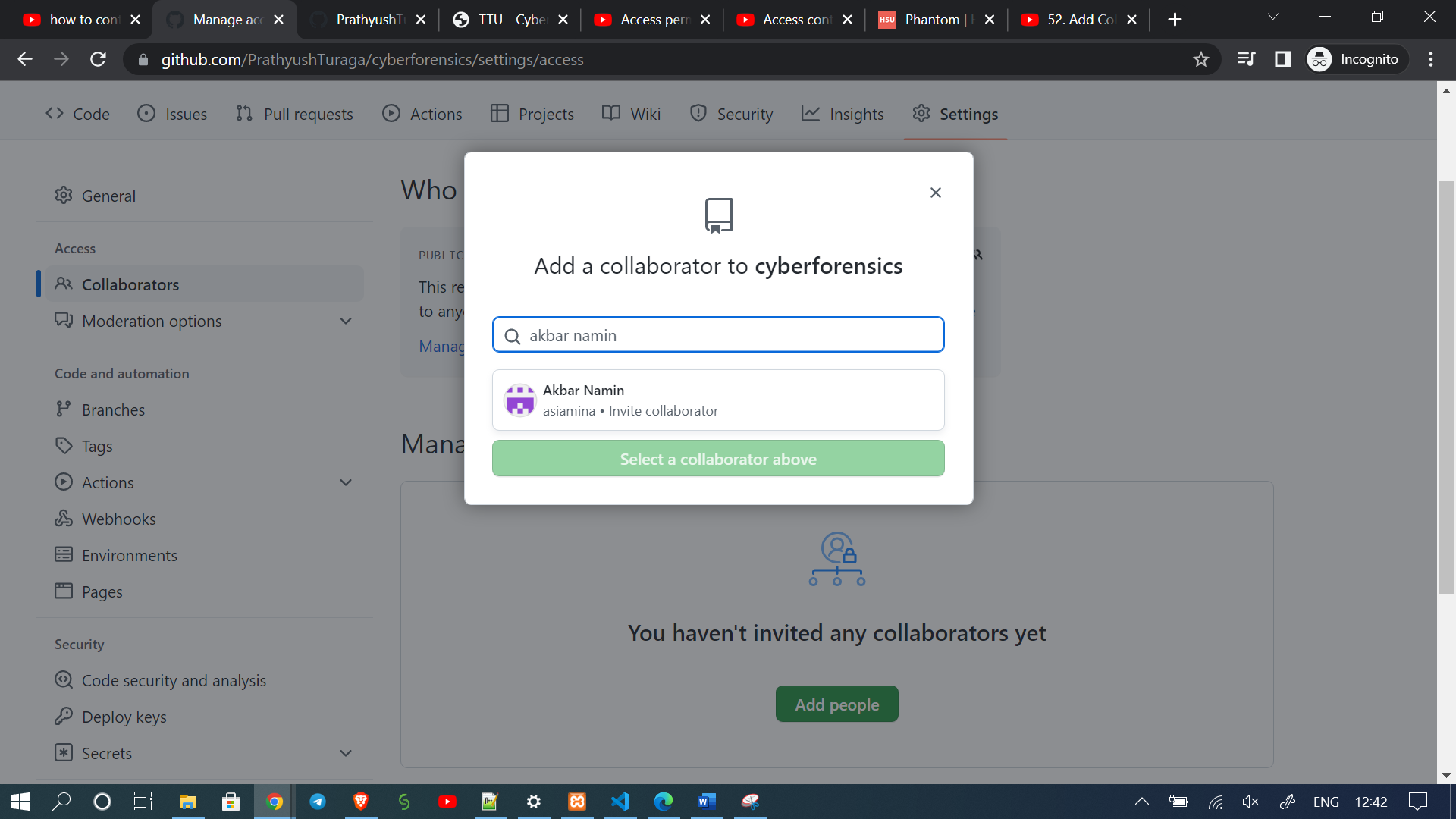
Go to Repository on GitHub -> Setting -> Danger Zone -> Change Repository Visibility





* 1. If we want to give permissions for a specific developer to edit the repository, we can follow collaborators approach as shown below.

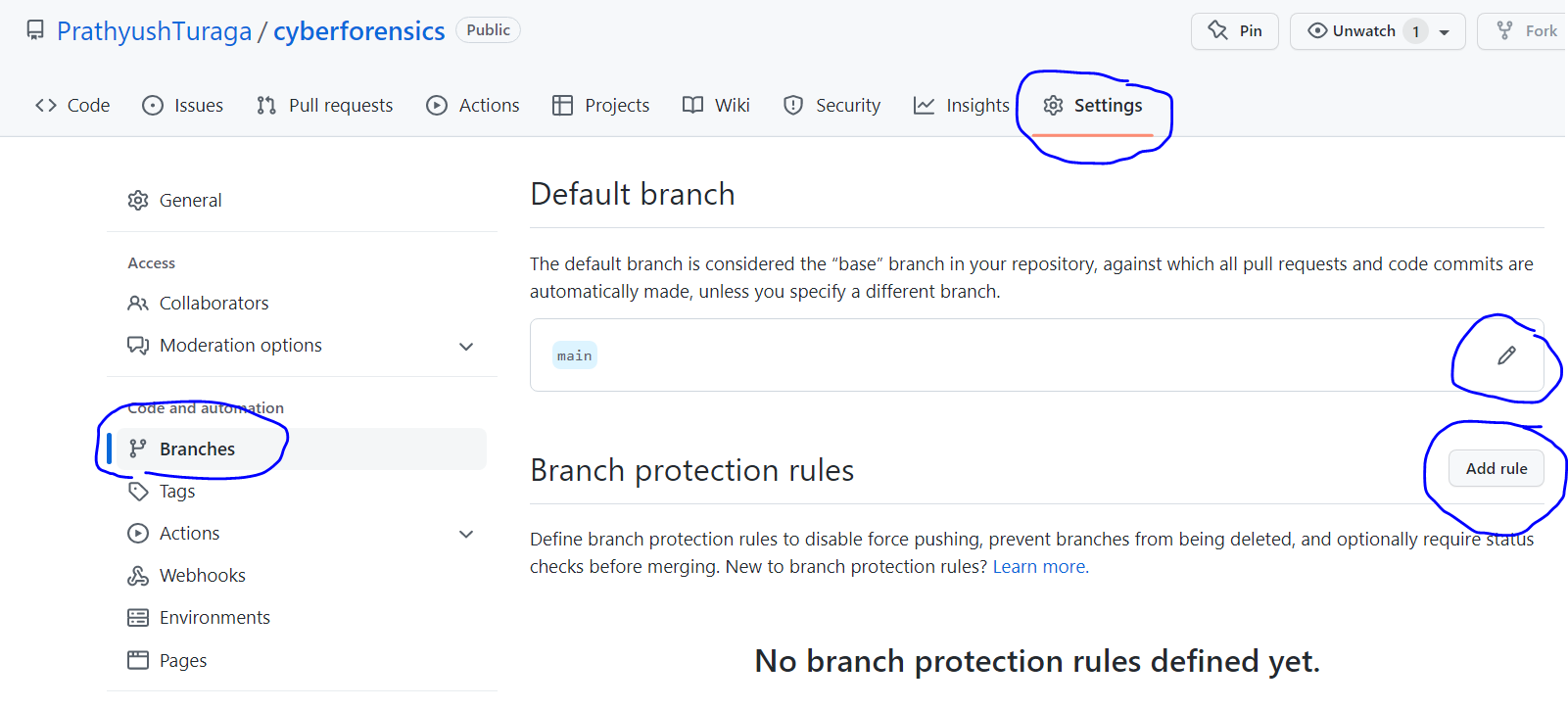
Go to Repository Settings -> Collaborators (Left side panel) -> Add People -> (Search for their GitHub ID and invite)



* 1. Adding Policy based rules for editing files.

For this feature, we can maintain different branches of Code (Dev, Test, UAT, Main) and maintain different access policies.

Go to Repository Settings -> Branches -> (Select required Branch)-> Add Rule



Click on required check-boxes and setup rules for the required branch. Below are the options.

