****Tarlac State University

**COLLEGE OF COMPUTER STUDIES**

Case Study

in

Integrative Programming Technology 2

Submitted By:

**Student 1**

**Student 2**

**Student 3**

Submitted To:

**Subject Teacher**

Date: []

Table of Contents

[A. Definition 3](#_Toc19107023)

[B. Case Analysis #1 4](#_Toc19107024)

[B.1 Problem 4](#_Toc19107025)

[B.2 Solution/Implementation 4](#_Toc19107026)

[B.3 Explanation 4](#_Toc19107027)

[C. Case Analysis #2 5](#_Toc19107028)

[C.1 Problem 5](#_Toc19107029)

[C.2 Solution/Implementation 5](#_Toc19107030)

[C.3 Explanation 5](#_Toc19107031)

[D. References 6](#_Toc19107032)

# Definition

The Lab Quiz#2 you did (Static Website) upload the documentation and files in your GitHub repository and add a minimum of 3 more webpages to add in your repository during the development use Git and GitHub and create a documentation of all.

**For the development of the Portfolio Webpages using (Git and GitHub) Issues, Pull Request, Milestone, Branching**

**Quiz # 1 LAB: Portfolio**

**Portfolio Web Pages Upload to GitHub Using Git**

**Requirements:**

* Each member in your group will need to upload **a webpage**
* Create a **Pull Request** for each member in Github with comments.
* Create an **Issue** for each member in Github and comment.
* Create a **Milestone** for the group in Github.
* Create a branch for each member and merge it with your master in Github.

**Define and describe how you use of each command with screenshot in your case study.**

* Git clone
* Git Pull
* Git Push
* Git Fetch
* Git Merge
* Issue
* Pull Request
* Milestone
* Branch

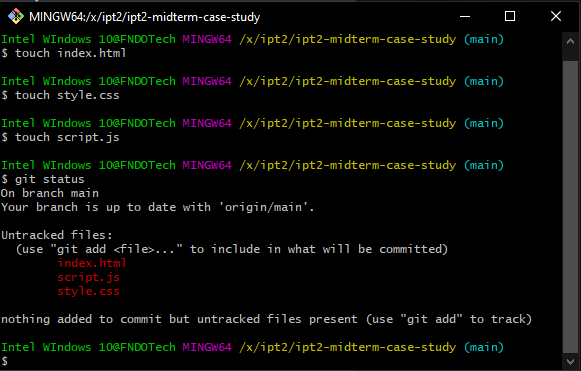
**Note: List the contributions of your group members**

# Case Analysis (Git and GitHub Workflow)

## Documentation

Create a narrative documentation, take screenshots to tell and proof your case study. List the things you have done and how you did to the case study, the commands you have use with screenshots. Include **git log** screenshot as reference of history of your commit in your documentation.

// creating a html, css, and js file to initialize the repository.  
// checking the status of the directory if there’s an issue and found that there are 3 untracked files

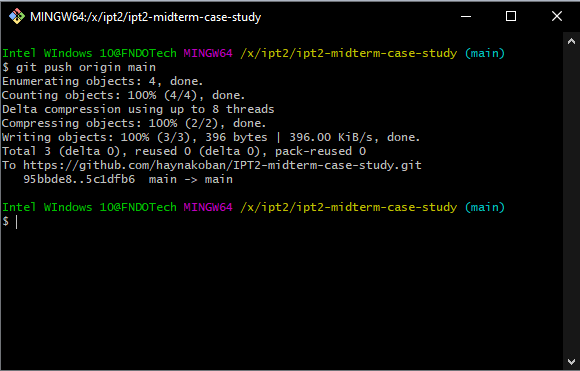


// to remove the untracked files, use “git add .” to stage the files  
// to commit the files, use “git commit” command

Text

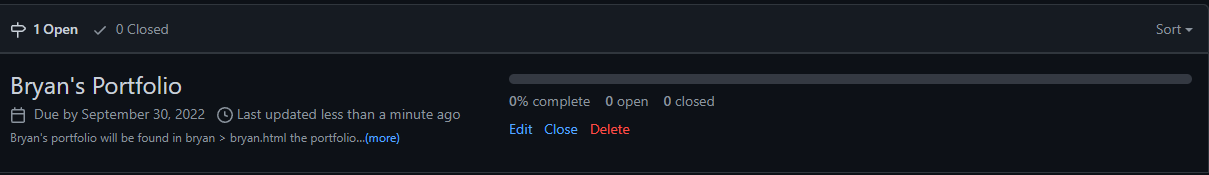
Description automatically generated

// pushing the initial state of the repository



## Bryan Cortez

// milestone for creating Bryan’s portfolio

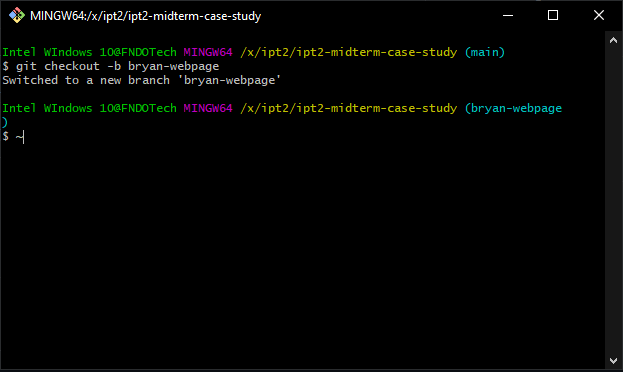


// cloning the GitHub repository

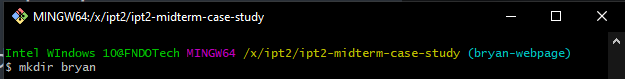
Text

Description automatically generated

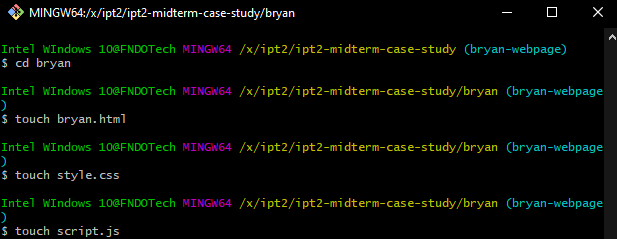
// create new branch



// create new directory

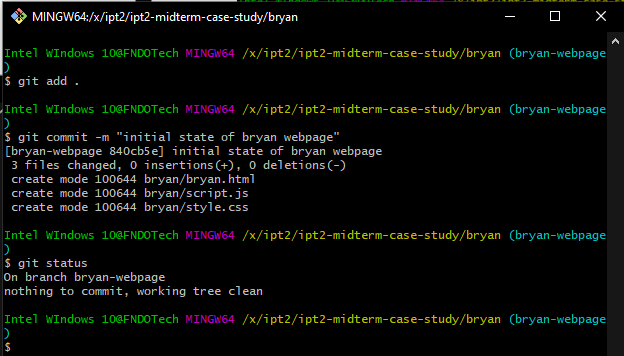


// changing directory and create new html, css, and js file



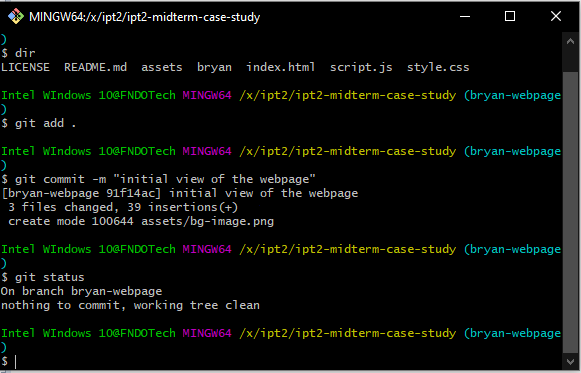
// staging the new files and committing it.

// on the branch bryan-webpage, the directory is clean and ready to push in GitHub

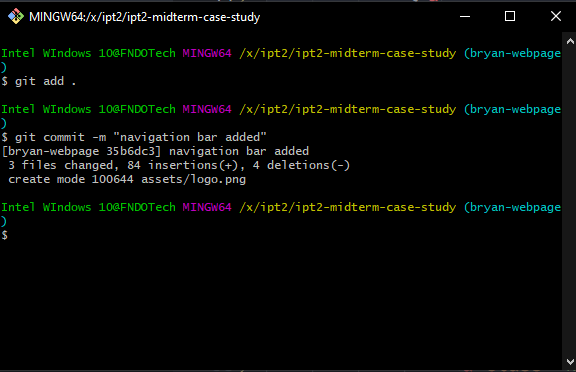


// new folder created assets

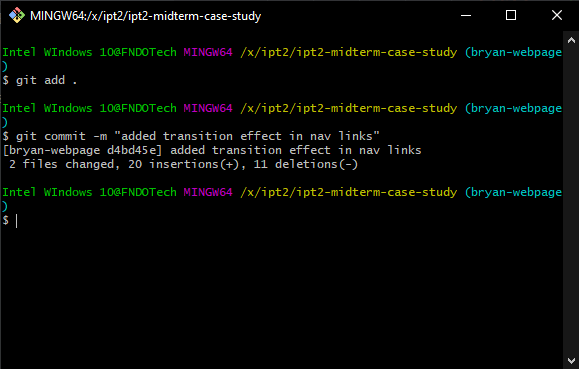
// staging and committing all the changes on branch bryan-webpage



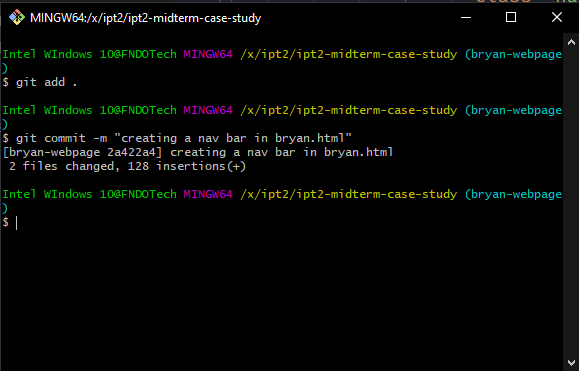
// staging and committing all the changes on branch bryan-webpage



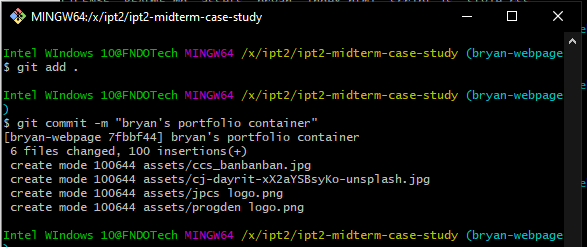
// staging and committing all the changes on branch bryan-webpage

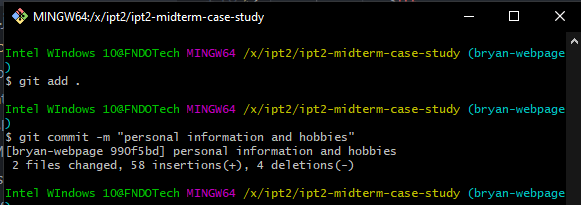


// now designing my portfolio

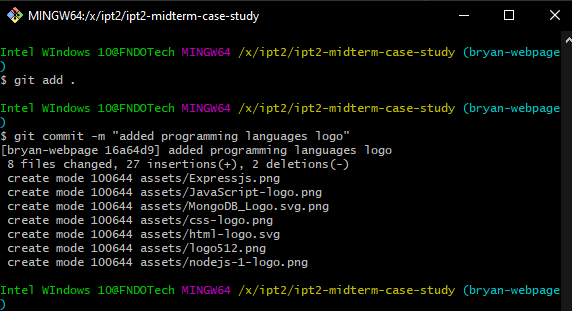


// stage and commit the container of my portfolio

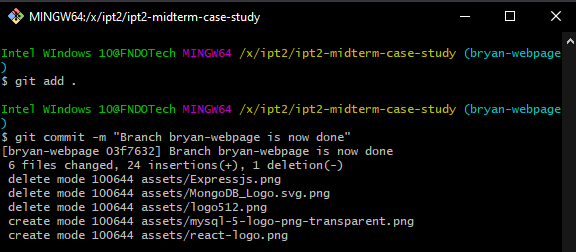
// stage and commit the changes. Finish the personal information and hobbies



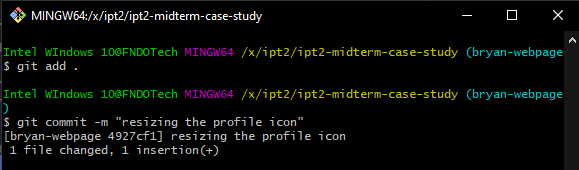
// stage and commit the logo of programming languages



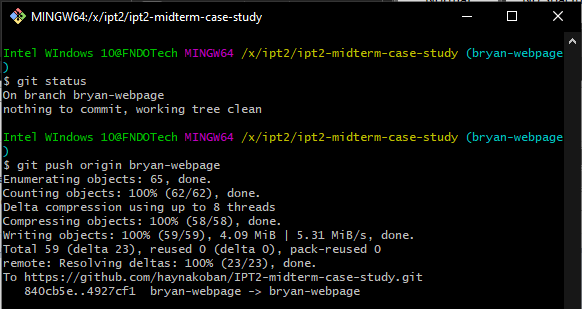
// stage and commit the last part of the portfolio

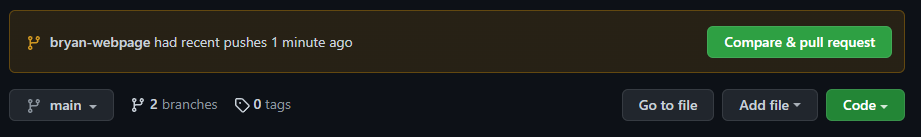


// stage and commit the resize icon

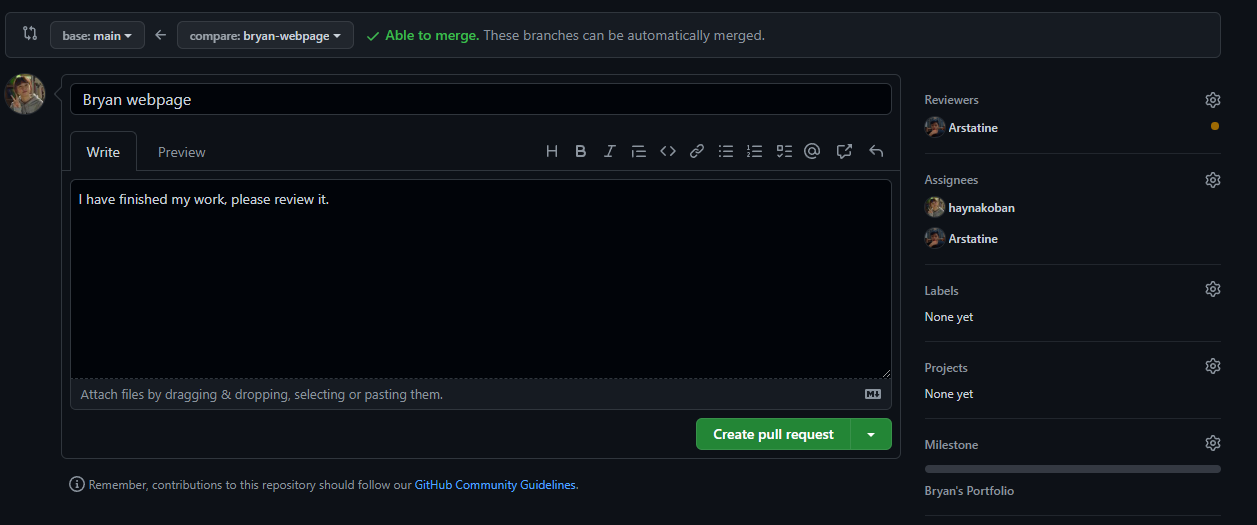


// pushing the changes on branch bryan-webpage

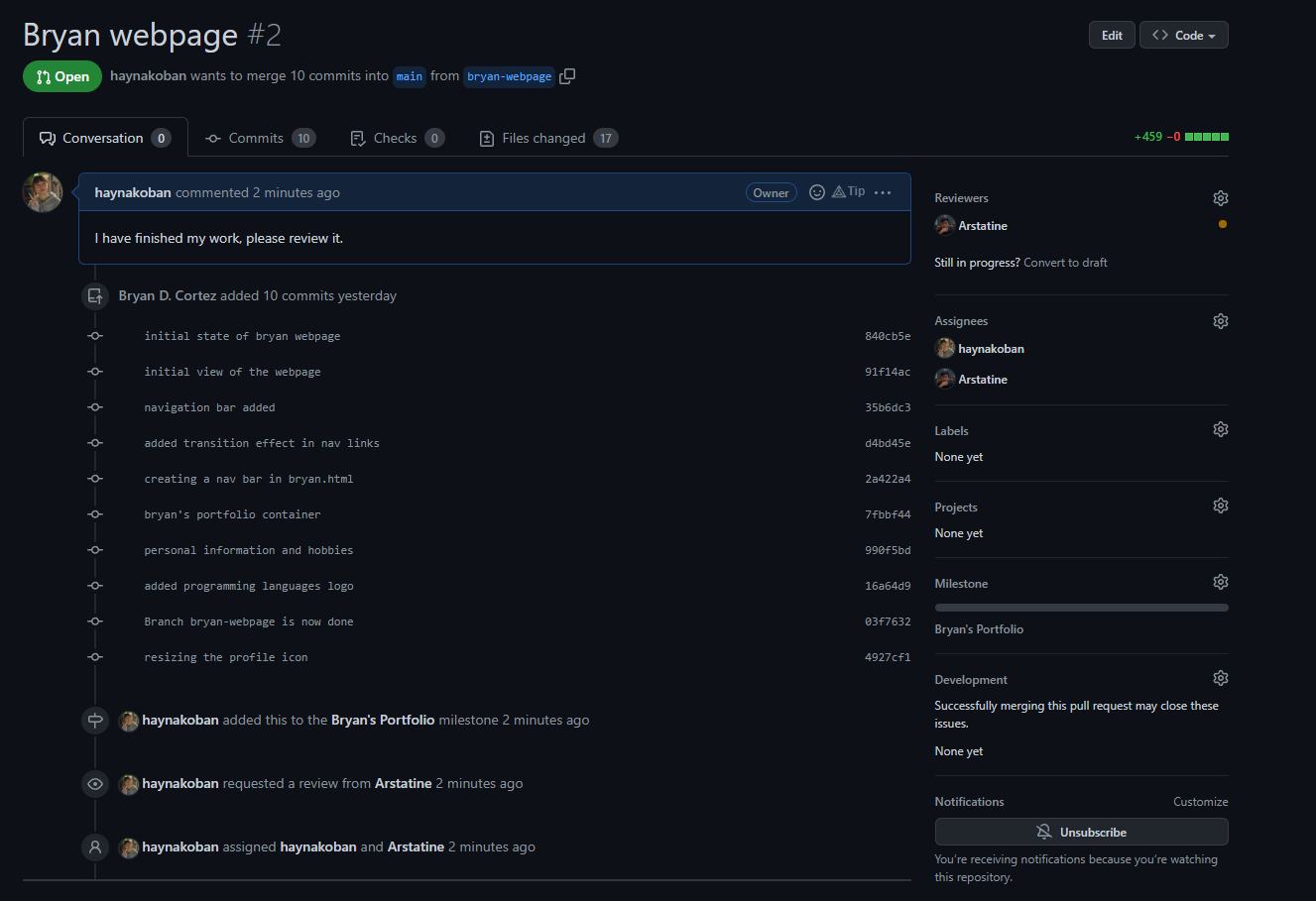


// compare and create a pull request

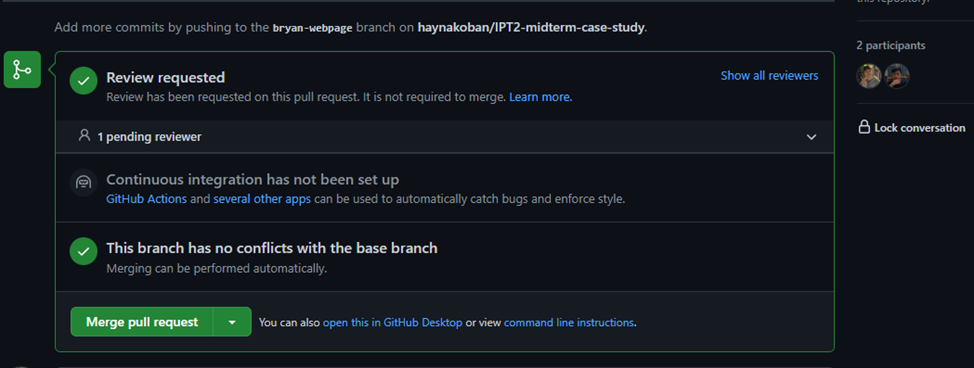
// filling the pull request



// the list of commits on the branch bryan-webpage



// merging the branch bryan-webpage to main



// pull request successfully merged and closed

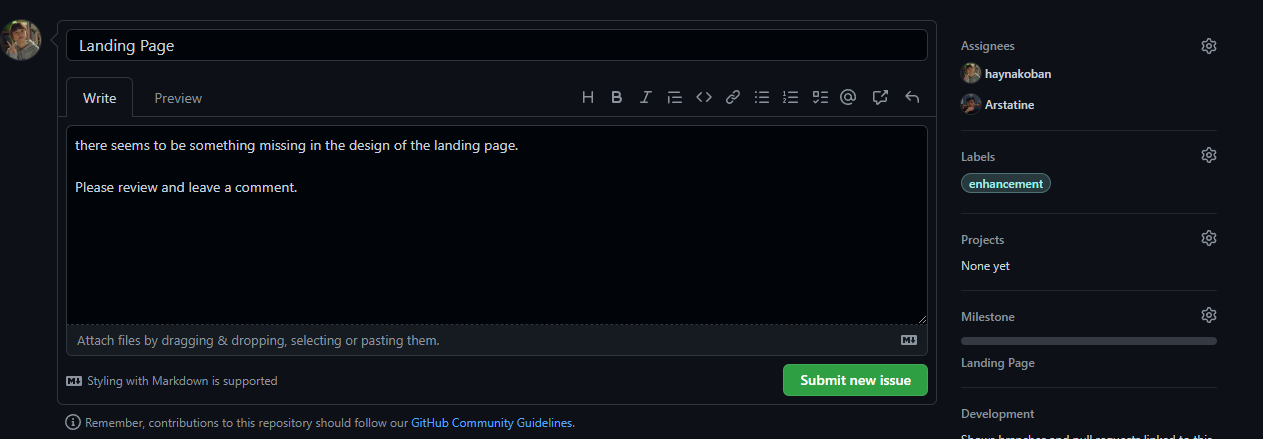
Graphical user interface, text

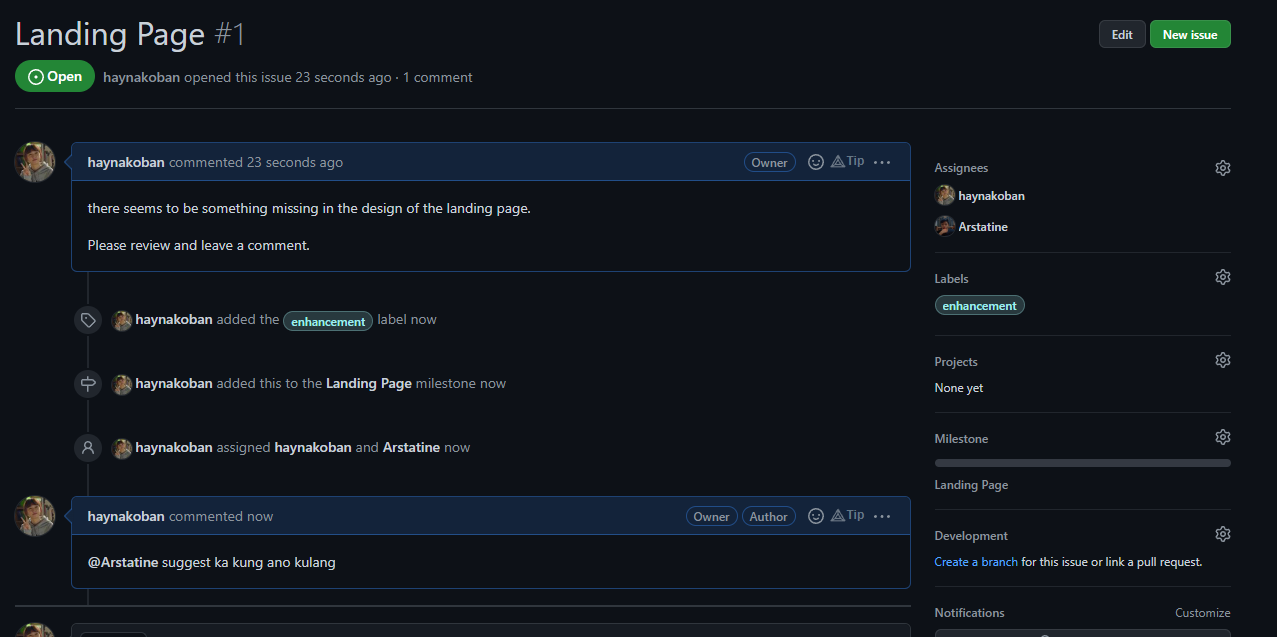
Description automatically generated

// deleted the bryan-webpage branch

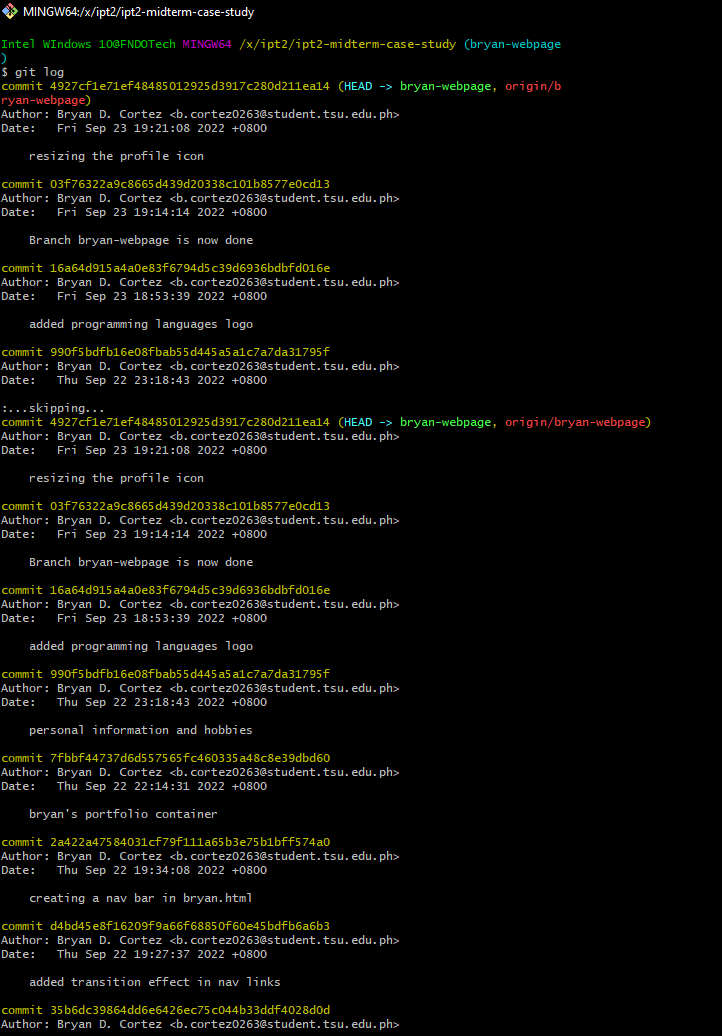


// creating an issue for the landing page





// use git log command to display the list of commits



// continuation of git log



// checking the new meta data that currently pushed in main branch

// and fetching the changes in my local git

Text

Description automatically generated

// pulling the new files or changes into my local git

Text

Description automatically generated

# References

Use IEEE format.

**Grade Matrix:**

Git Command : **20%**

GitHub Implement : **20%**

Documentation : **40%**

Webpage (Development) : **30%**

**100%**