**CIE203 Software Engineering 2016 Lab 8: : Requirements (1 marks)**

**What is GitHub?**

GitHub is a code hosting platform for version control and collaboration. It lets you and others work together on projects from anywhere.

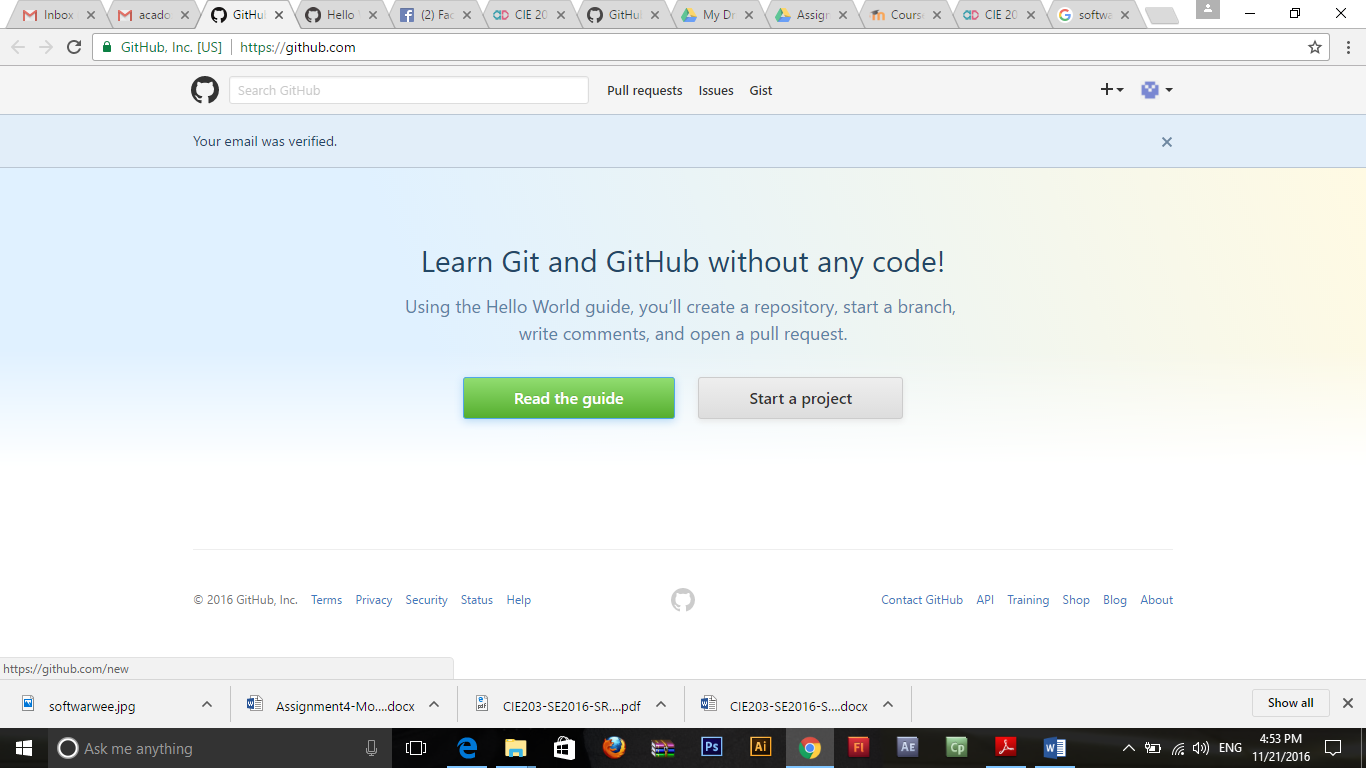
# Objectives

1. Using Githup to manage shared projects.

Create your account in <https://github.com/>

## Step 1. Create a Repository

A **repository** is usually used to organize a single project. Repositories can contain folders and files, images, videos, spreadsheets, and data sets – anything your project needs.



### To create a new repository

1. In the upper right corner, next to your avatar or identicon, click  and then select **New repository**.
2. Name your repository hello-world.
3. Write a short description.
4. Select **Initialize this repository with a README**.



## Step 2. Create a Branch

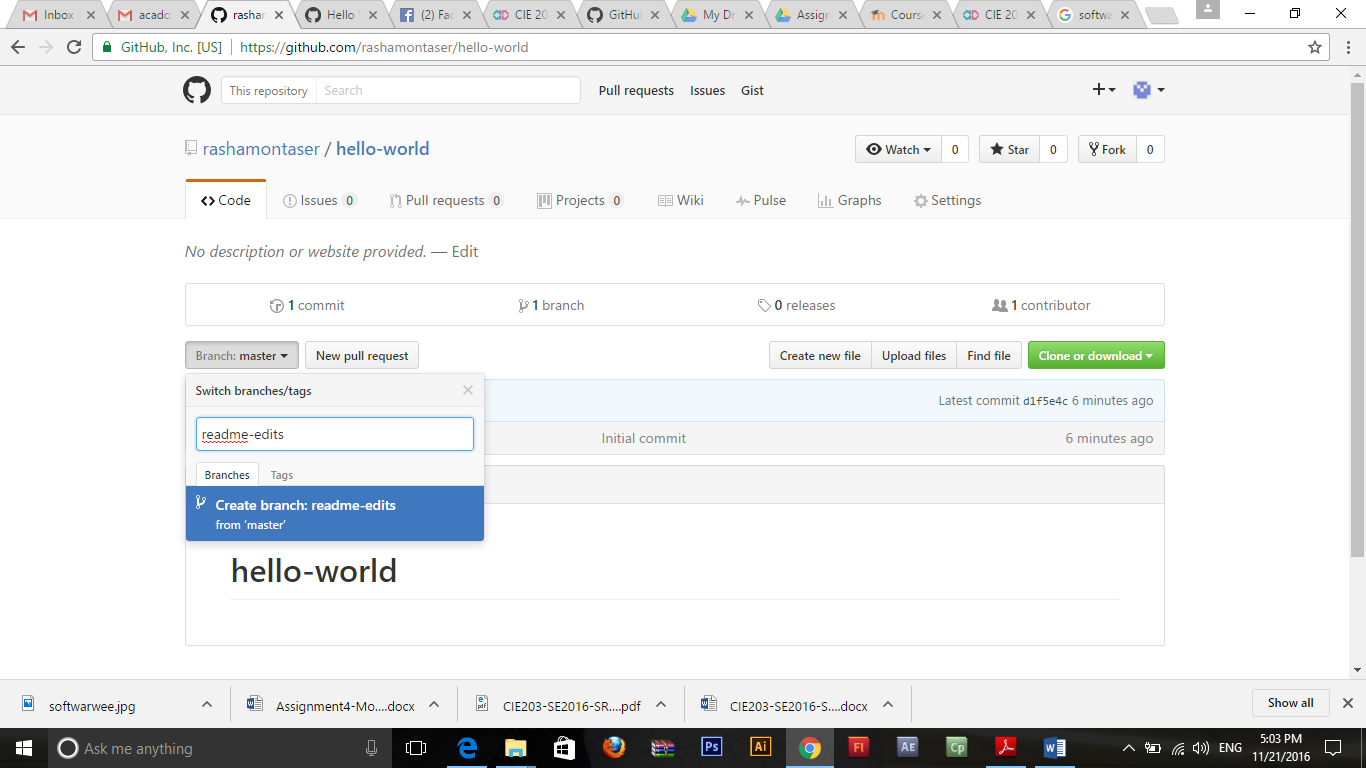
**Branching** is the way to work on different versions of a repository at one time.

By default your repository has one branch named master which is considered to be the definitive branch. We use branches to experiment and make edits before committing them to master. If someone else made changes to the master branch while you were working on your branch, you could pull in those updates.

Branches are used for keeping bug fixes and feature work separate from master (production) branch. When a change is ready, they merge their branch into master.

### To create a new branch

1. Go to your new repository hello-world.
2. Click the drop down at the top of the file list that says **branch: master**.
3. Type a branch name, readme-edits, into the new branch text box.
4. Select the blue **Create branch** box or hit “Enter” on your keyboard.



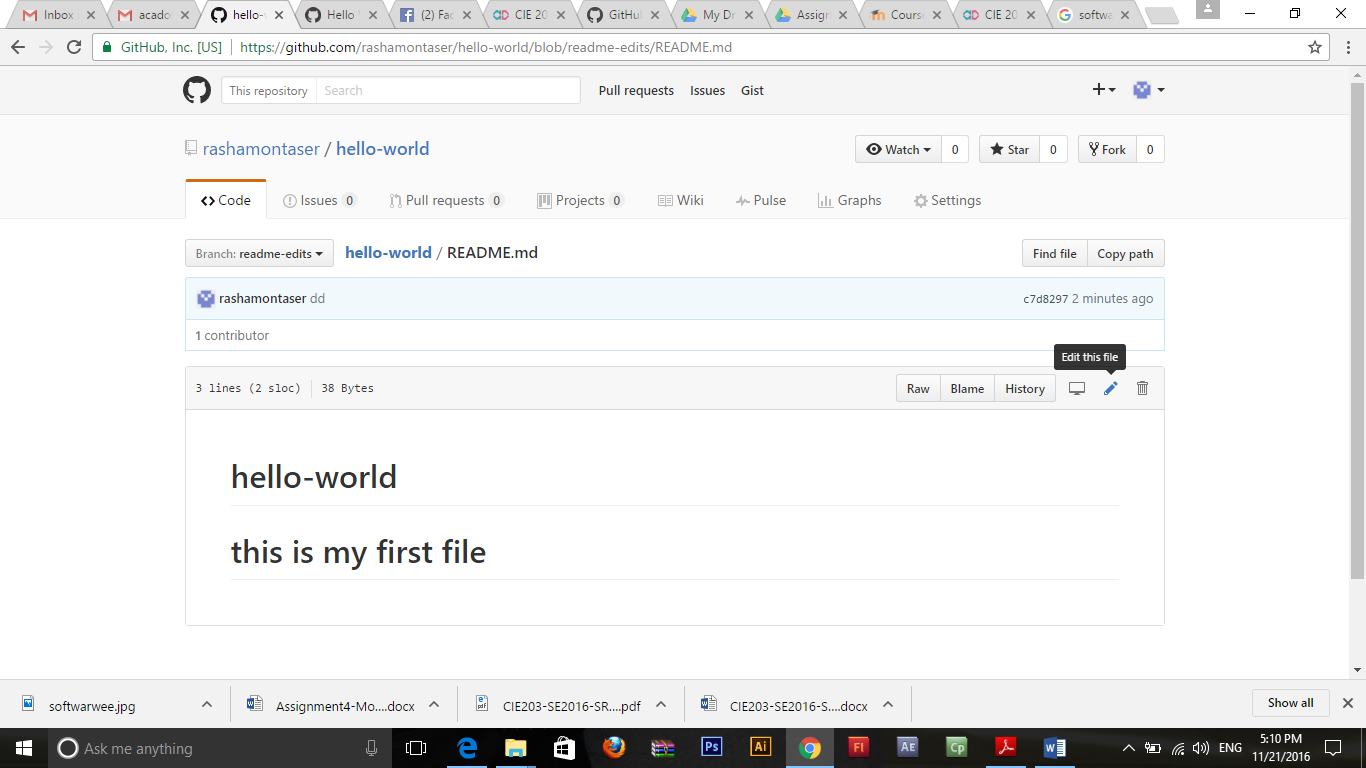
Now you have two branches, master and readme-edits. They look exactly the same, but not for long! Next we’ll add our changes to the new branch.

## Step 3. Make and commit changes

On GitHub, saved changes are called commits. Each commit has an associated commit message, which is a description explaining why a particular change was made. Commit messages capture the history of your changes, so other contributors can understand what you’ve done and why.

#### Make and commit changes

1. Click the README.md file.
2. Click the  pencil icon in the upper right corner of the file view to edit.
3. In the editor, write a bit about yourself.
4. Write a commit message that describes your changes.
5. Click **Commit changes** button.





These changes will be made to just the README file on your readme-editsbranch, so now this branch contains content that’s different from master.

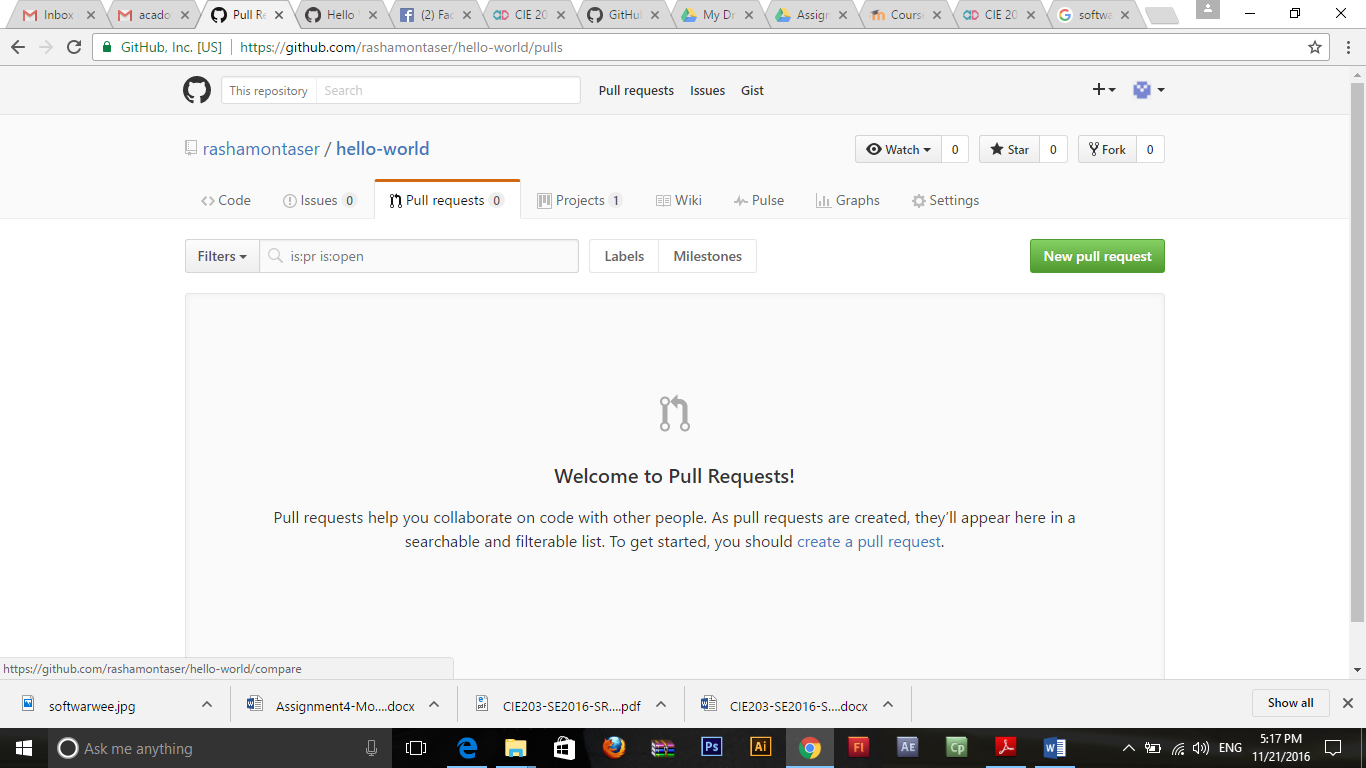
## Step 4. Open a Pull Request

Now that you have changes in a branch off of master, you can open a pull request.

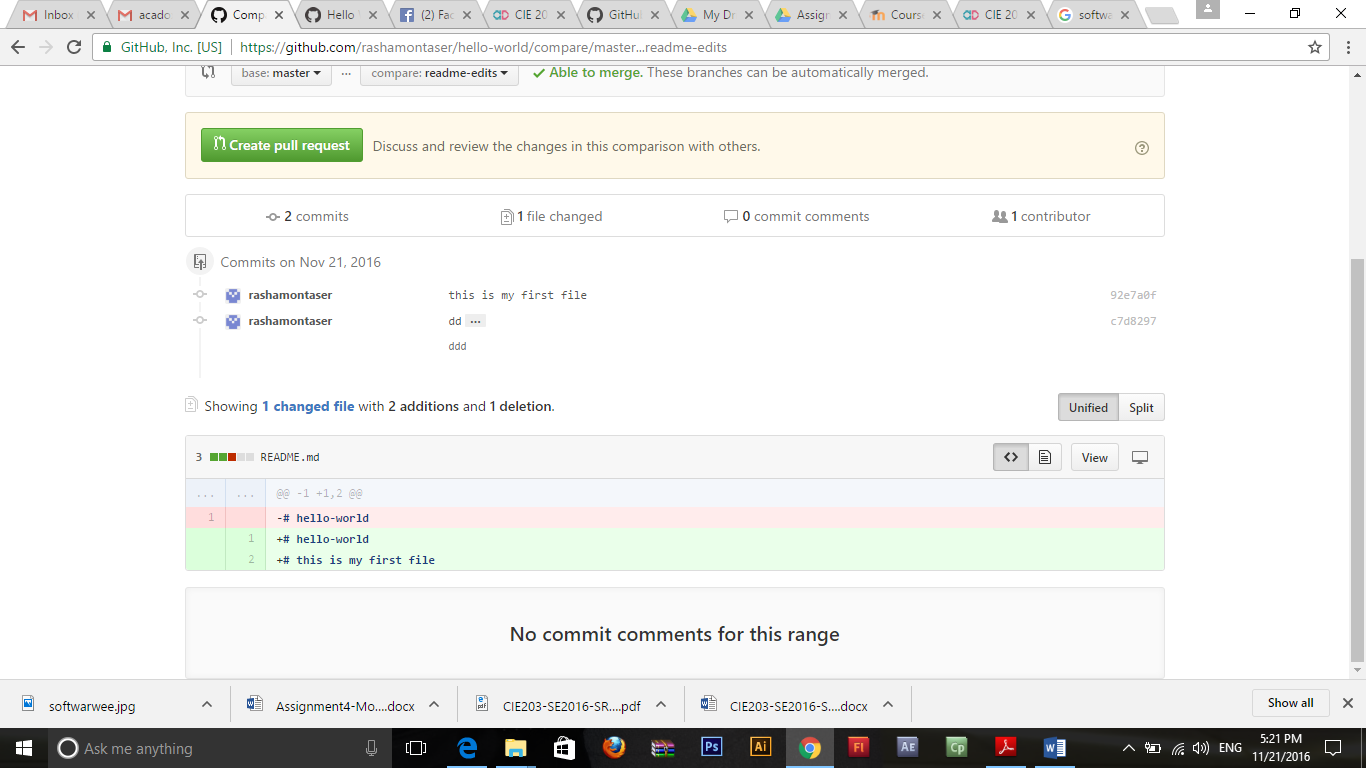
When you open a pull request, you’re proposing your changes and requesting that someone review and pull in your contribution and merge them into their branch. Pull requests show diffs, or differences, of the content from both branches. The changes, additions, and subtractions are shown in green and red.

#### Open a Pull Request for changes to the README

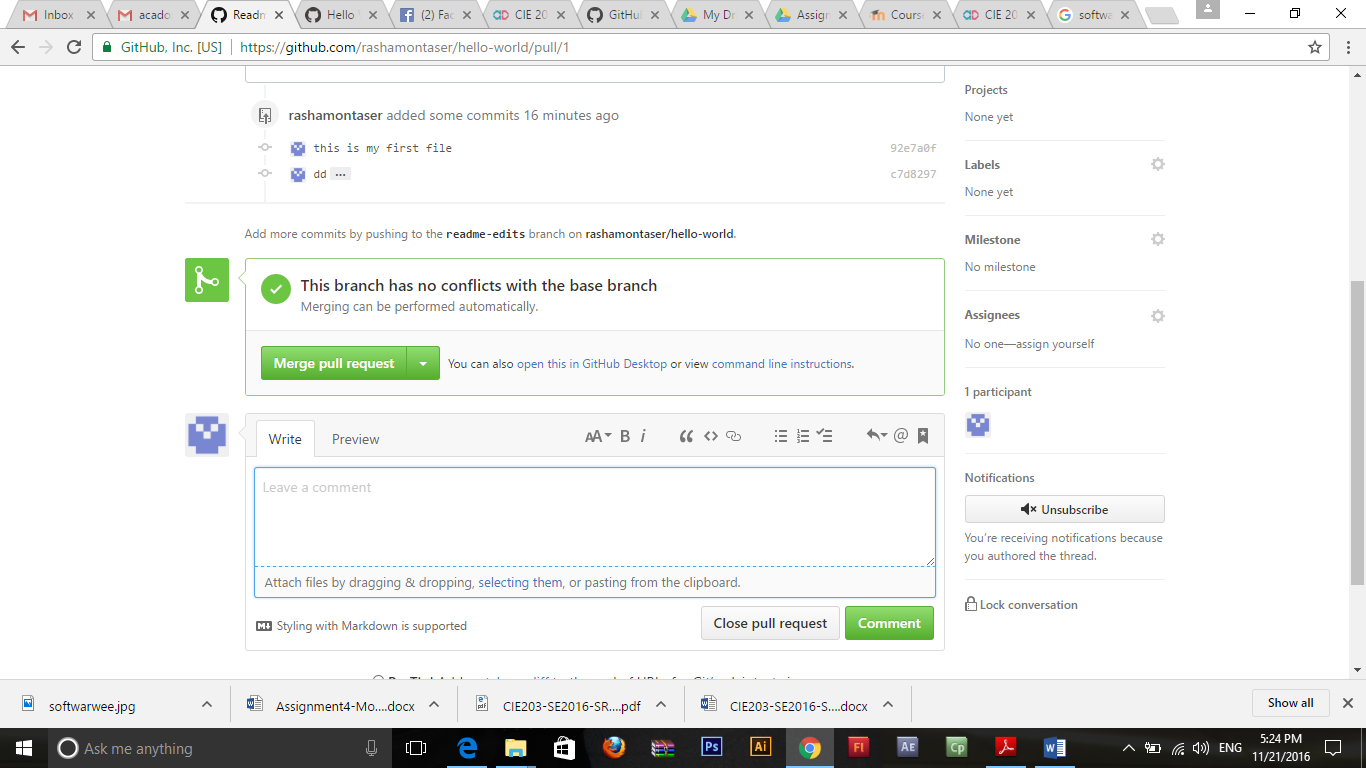
1. Click the  **Pull Request** tab, then from the Pull Request page, click the green **New pull request** button.



1. Select the branch you made, readme-edits, to compare with master (the original).
2. Look over your changes in the diffs on the Compare page, make sure they’re what you want to submit.
3. When you’re satisfied that these are the changes you want to submit, click the big green **Create Pull Request**button.



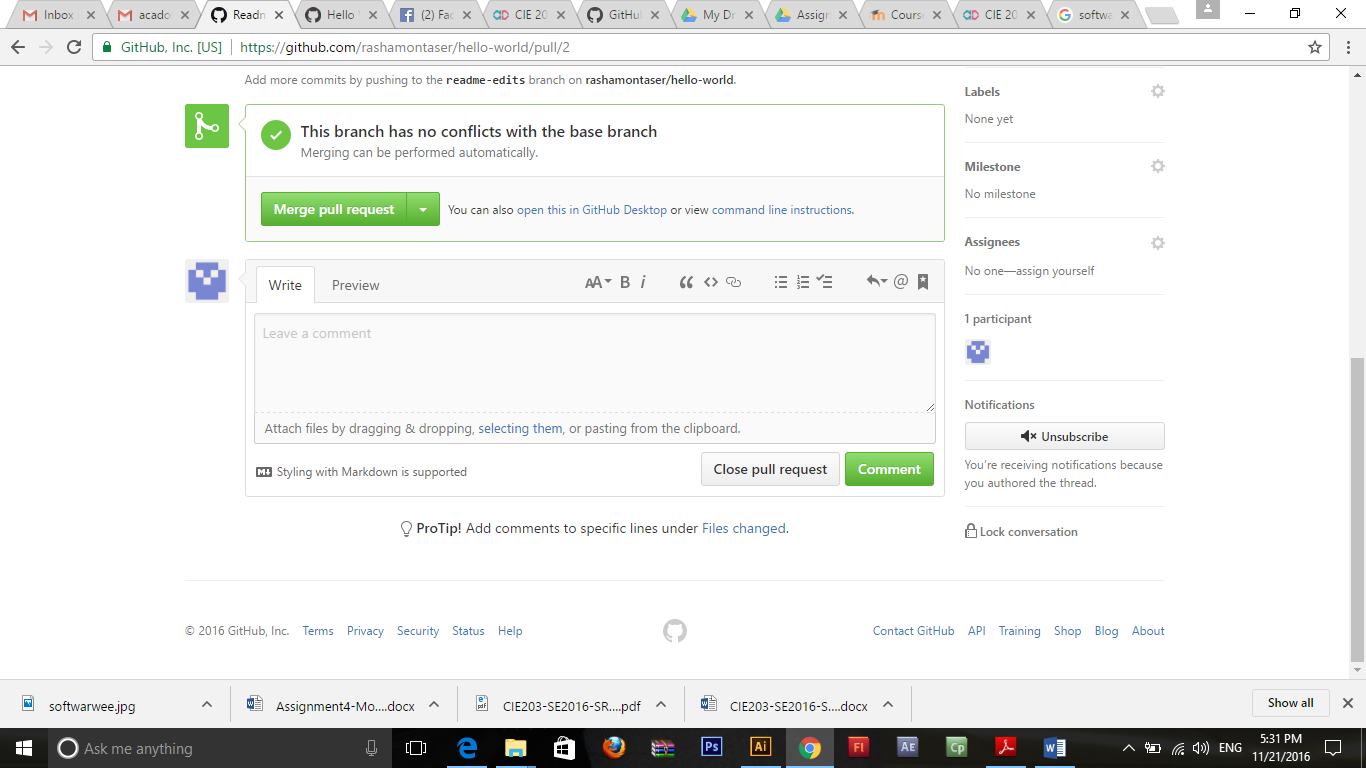
1. Give your pull request a title and write a brief description of your changes.



## Step 5. Merge your Pull Request

In this final step, it’s time to bring your changes together – merging your readme-edits branch into the master branch.

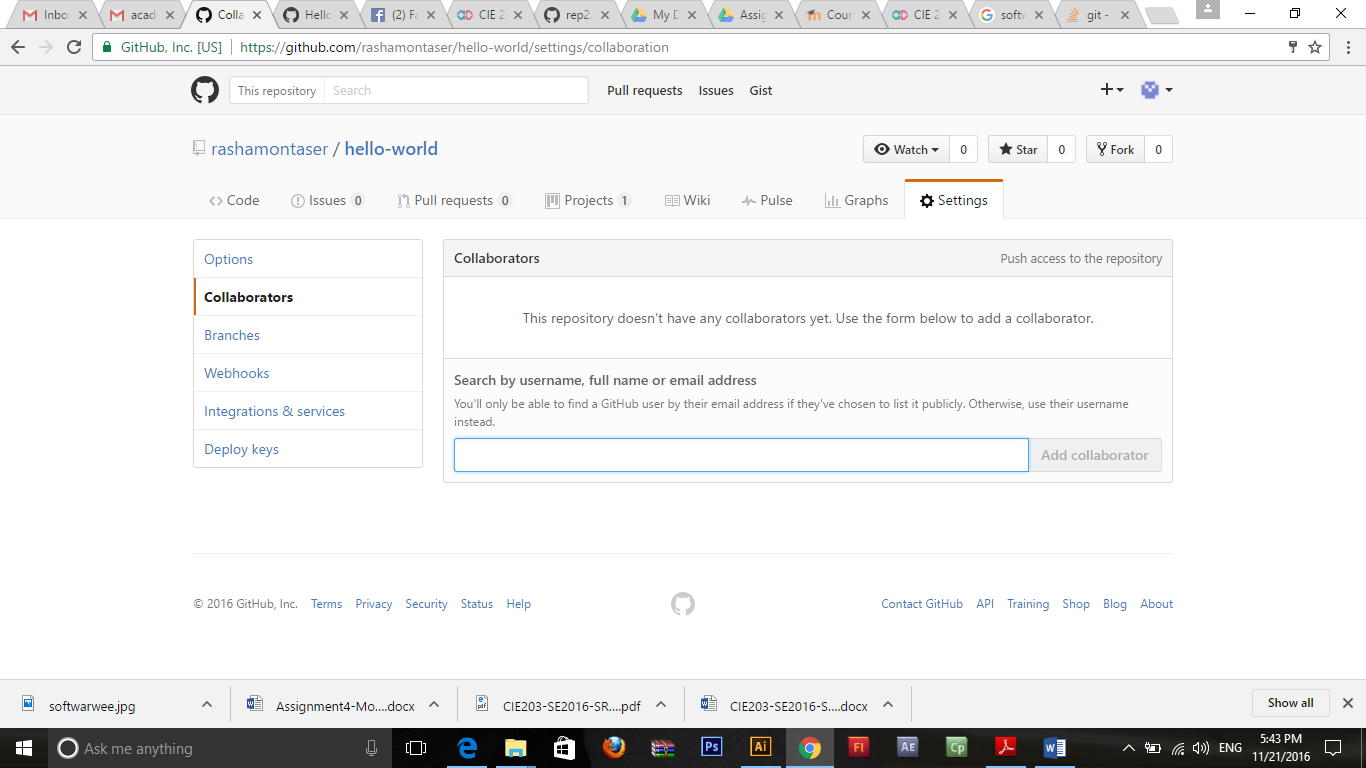
1. Click the green **Merge pull request** button to merge the changes into master.
2. Click **Confirm merge**.
3. Go ahead and delete the branch, since its changes have been incorporated, with the **Delete branch** button in the purple box.



## Step6: Add Collaborators to your project

Collaborators are your team members who access and add files, comments in your project.

1. Navigate to the repository on Github you wish to share with your collaborator.
2. Click on the "Settings" link in the right side menu, below "Network"
3. On the new page, click the "Collaborators" menu item on the left side of the page.
4. Start typing the new collaborator's GitHub username into the text box.
5. Select the GitHub user from the list that appears below the text box.
6. Click the "Add" button.



Student Activity #1

Listen to the following videos, then create account on <https://github.com/> , upload your SDD and allow your team members to work with your created project.

<https://www.youtube.com/watch?v=iIWM5k2d4cs>

<https://www.youtube.com/watch?v=dZh2U5g2NMs>

<https://www.youtube.com/watch?v=hFSaDki19HI>