Assignment #4

Due Date: 4/22/16 by 11:59pm

Deliverable: post your homework as a SINGLE PDF file on Blackboard with the name "HW4_YourLastName"

The Assignment is composed of two parts: **Part-A** and **Part-B**

Part-A:

- Consider the data listed in the following matrix for a product of size 90 KLOC:
  - Calculate the defect removal rate for every phase
  - Calculate the defect injection rate for every phase
  - Calculate the defect escape rate for every phase
  - Which phase is the most effective in removing defects?
  - Calculate the overall defect removal effectiveness.
  - Do you think reviews and inspections were effective? Explain.
  - If the number of defects originated in requirements phase increased by 25% and defects detected in requirements review increased by 25%, do you think that will have a positive or negative impact on the defects originated in the coding phase? Explain your answer in detail (present data to support your answer).
  - If the number of defects originated in design phase increased by 5% and defects (defects escaped from prior phases and injected in current) detected in code inspections increased by 95%, do you think that will have a positive or negative impact on defect removal effectiveness for the testing phases? Explain your answer in detail (present data to support your answer).

		Defect Origin							
		Requirement	Analysis	Design	Coding	Unit Testing	Integration Testing	System Testing	Field
Where Found	Requirement	88							
	Analysis	110	55						
	Design	67	59	99					
	Coding	34	63	72	311				
	Unit Testing	44	44	55	121	10			
	Integration Testing	34	9	32	38	-	11		
	System Testing	6	5	5	23	-	-	9	
	Field	7	2	4	12	-	-	-	3

Part-B:

High-Level Requirements:

1. **Part-B** of the assignment will be completed in 2 phases.
2. All your deliverables will be on GitHub.
3. Assignment #4 Part 2, Phase I will complete on 4/14/2016 and Phase II will complete on 4/21/2016 by 11:59pm

Phase I Requirements:

1. You have to create your GitHub account id as instructed in the tutorial.

- For a student with the name John Doe with A number A12345678.
- His id will be JSP16SCM78D.
- In this ID J is the first character of his first name.
- SP16SCM will be common for each student.
- 78 is the last two numbers of his A number.
- D is the first character of his last name.

Once the TA receives all your GitHub accounts you will be added as Collaborators to the master repository. For this the TA will be sending out a mail to get all your created GitHub accounts. Use your hawk id while registering for your GitHub accounts. Do not use newly created email ids just for the assignment as GitHub may consider these id's as Bots and remove them. You will be proceeding to the next step only after the TA sends out a mail confirming that you have been added as a collaborator.

2. Next you will be forking the repository SP16SCM in the GitHub account sp2016scm to the GitHub account that you have created. This repository can be opened with the following URL.
https://github.com/nithinrajeev/SP16SCM.git

3. Now using the URL given above you will login to the repository listed above. You will go to the issues tab and create a new issue. The title of

the issue will be as listed below. This issue will be labeled as an enhancement.

"<Your GitHub id> adding first name"
Example – "JSP16SCM adding first name"

4. You will now clone the repository that you have forked into your account on to your local machine. This repository will contain files a.java, b.java, c.java, …, z.java. You will open the file that has the first character of your first name. Once you open the file you will enter your GitHub account id in a new line and save it.

   Example – For John Doe with GitHub account JSP16SCM78D, he will open j.java and add his GitHub account username in a new line and save it. So j.java will contain JSP16SCM78D.

5. Now you will do a commit and then push this repository into your GitHub account. While making the commit you are to add your Issue number in the commit message in the following format.

   Example – git commit –m "Fixes #<IssueNumber>. <GitHub username> first name added."

   Say for John Doe the issue he has created is Issue #24. His commit message will be as follows.

   git commit –m "Fixes #24. JSP16SCM78D first name added"

6. Once that is done you will create a pull request to the master repository so that your changes can be merged onto the master repository. You will just create the pull request. It will be the TA who will accept your changes. Your pull request title should also be the exact same as your commit message.

   This will complete your Phase 1. You will also need to submit two screenshots as the commit report for each phase. The last page in this document instructs you on how to capture those screen shots. Both

the screenshots that you have captured need to be put in a pdf called Report_Phase1_<GitHubUsername>.pdf for your phase I. Once your changes are all accepted in the master repository you will receive an email to commence Phase II. Then you will be following the Phase II instructions.

Phase II Requirements:

1. You will open the master repository from your GitHub account and create a new issue. This issues title will be as listed below. This issue will be labeled as a bug.

   "<Your GitHub id> adding last name"
   Example – "JSP16SCM adding last name"

2. Now you will go to the forked repository in your account and update the repository. This is done using the following command.

   git remote add upstream <Master Repository Git URL>
   git fetch upstream
   git rebase upstream/master

3. Update your local clone to the latest in your repository.

4. Now you will open the file with the first character of your last name. Then you will be adding your GitHub account username in a new line

   Example – John Doe will be opening the file d.java and add his GitHub account username which is JSP16SCM78D in a new line.

5. Now you will do a commit and then push this repository into your GitHub account. While making the commit you are to add your Issue number in the commit message in the following format.

   Example – git commit –m "Fixes #<IssueNumber>. <GitHub username> first name added."

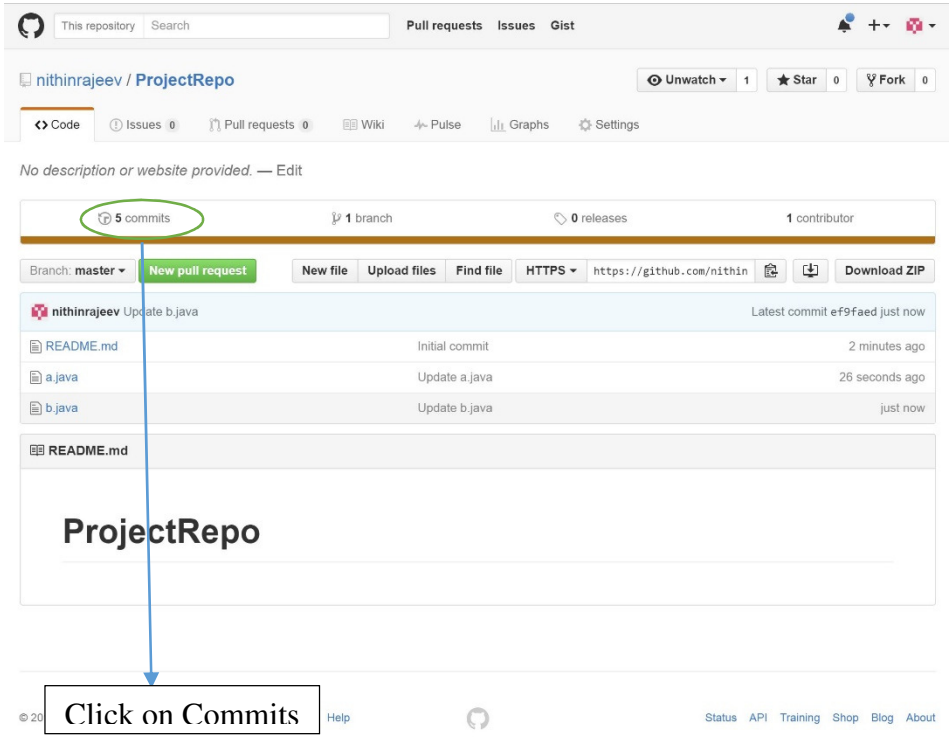
Say for John Doe the issue he has created is Issue #65. His commit message will be as follows.
git commit –m “Fixes #65. JSP16SCM78D last name added”

6. Once that is done you will create a pull request to the master repository so that your changes can be merged onto the master repository. You will just create the pull request. It will be the TA who will accept your changes. Your pull request title should also be the exact same as your commit message.

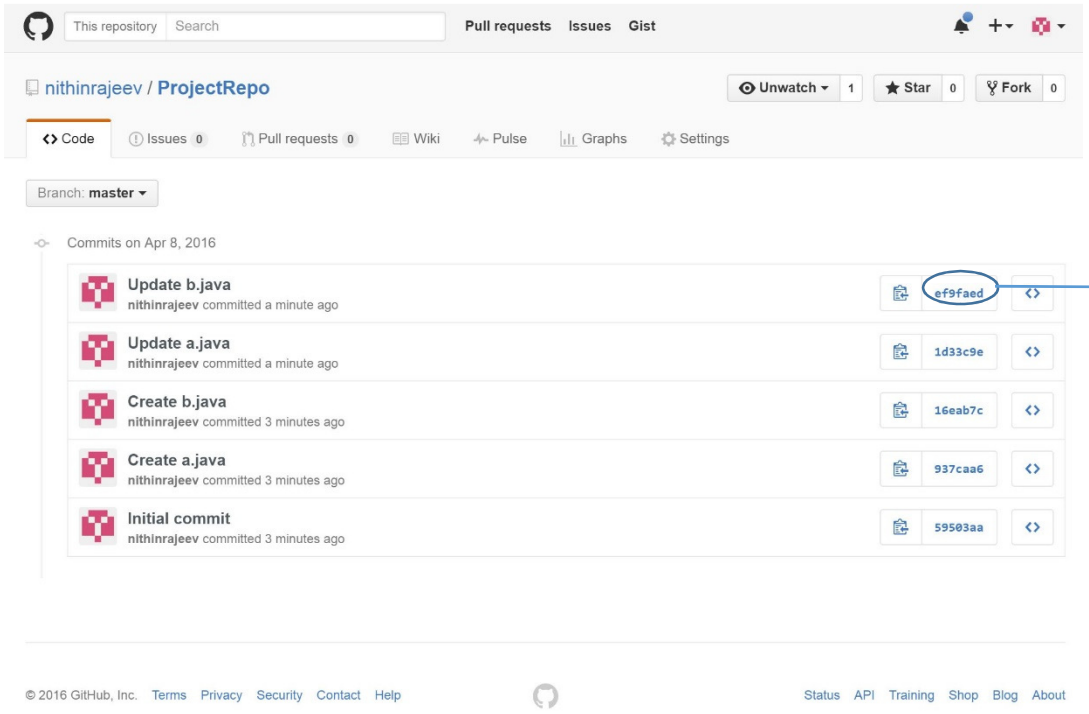
   This will complete your Phase II. You have to capture the same screenshots mentioned above for phase II and attach it in a document called Report_PhaseII_<GitHubUsername>.pdf and submit it.

Deliverable for Part-B :

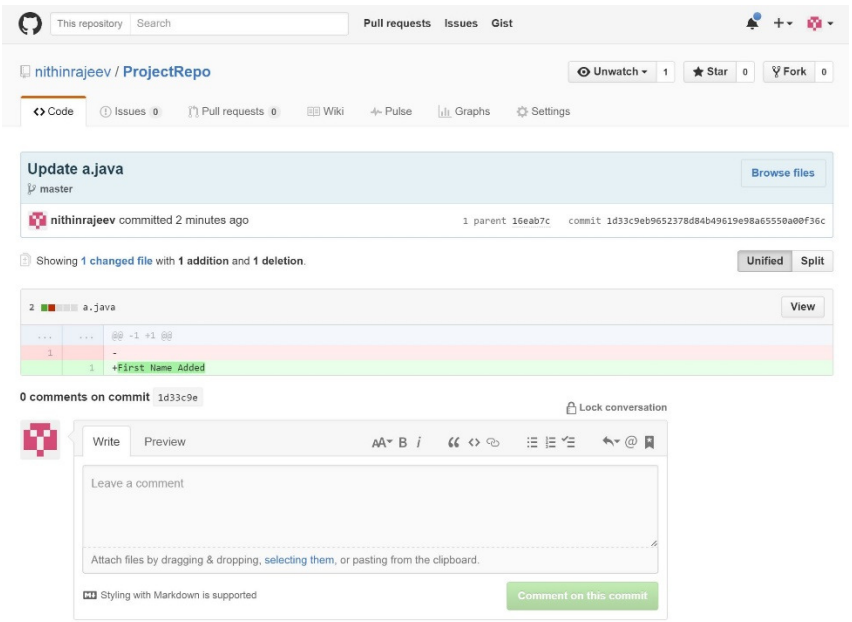
1. Login to your GitHub account and open the repository that you have committed to. Click on the commits Tab.



2. You will get the following page. You are to take a screenshot of this page. Click on the commit that you have just made to expand it.



3. This will expand and show you the individual commit. You have to take a screenshot of this page as well.



The report that you have created will be added to the report folder in the repository and submitted. This means that for the phase I report, you will be adding the report document in that Phase1_Report folder and committing it. And for Phase II you will be adding the report document in the folder Phase2_Report folder and committing it. Make sure that the report for each phase has to be committed and pushed to your repository BEFORE you initiate a pull request.