**Handover documentation for detailed steps from creating branch to giving pull request to Test and solving conflicts on Master**

**-Rutuja Jori**

* **How to create a branch and push the code on your created branch:-**

1. Login to PHP JIRA (192.168.1.164:8080 Internal IP / 114.143.27.19:8080/http://15.207.46.183:8080/ External IP) with your username and password.
2. After successful login, select project from the **‘Projects’** tab for which you want to create issue or solve an issue assigned to you.
3. Creating a ticket/issue on JIRA
4. select issue type(If you are adding new feature, issue type will be ‘Improvement’, if you are solving an issue then issue type will be ‘Bug’)
5. Enter Title (If adding a new feature it will start with ‘Suggestion:’ and a short title what you are adding. If solving an issue it will start with ‘Functional:’ and a short title about the issue.)
6. Add label (Your name)
7. Attached file if You have (If solving an issue attach screenshot of that page. For a new feature if you have created any documentation then attach the same)
8. Select date (the date when you are creating the issue)
9. Enter description (Detailed description which will include the steps from login)
10. Select Assignee
11. Select Priority
12. Select Component
13. Click on create
14. Now open that JIRA ticket which was recently created/the issue which is assigned to you. On the right side at top you will see ‘Search’ option, in that insert the issue ID (For eg. SMC-5242)
15. Now you will see the 5th button at top after the title of the issue, ‘Start Progress’ so click on that and status will change to ‘In Progress’.
16. Now go down see in left with down side panel in development option
17. Click on create branch
18. After clicking the create branch. Copy the branch name and click on ‘Cancel’.
19. Login with bitbucket account user. (Note:- if you already logged in please ignore this step)
20. Select repository (i.e. the Project on which you are going to work on)
21. Click on the left side select 3rd option ‘Branches’. Then at right side you will get a button ‘Create Branch’ click on it.
22. Select issue type
23. Paste the branch name that was copied.
24. Click on create
25. Copy git fetch && git checkout --branch name—
26. Open the folder on your local system where the project is (for e.g. E:\xampp\htdocs\localhost\smartcookie)
27. Now right click, and click on ‘Git Bash Here’.
28. Paste the git fetch && git checkout --branch name— form step 10. And click on enter.
29. git pull
30. Do changes whatever you want and save file and test the same on your localhost for issue is resolved or not/ new functionality is proper working or not.
31. git status
32. git add file name (copy paste file path which we get after git status)
33. git status
34. git commit -m "SMC-5242 short description of what you did"
35. git push

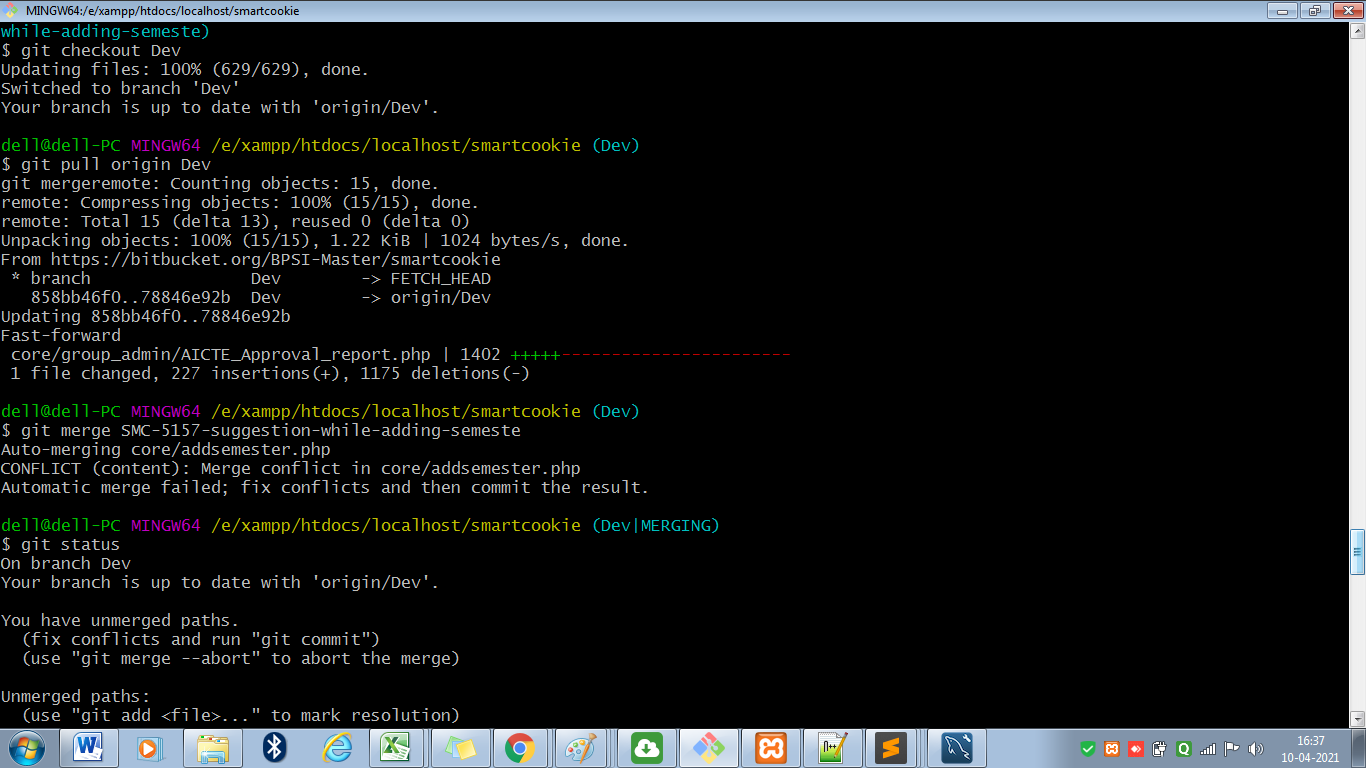
* **Merging code on Dev (for SmartCookie) and dev (for Startup World) :-**

1. After above step 20.
2. git checkout Dev/dev
3. git pull origin Dev/dev
4. git merge branch name
5. Now after merge there are 2 possibilities :
6. You will not get conflicts if a window opens as shown in below image:



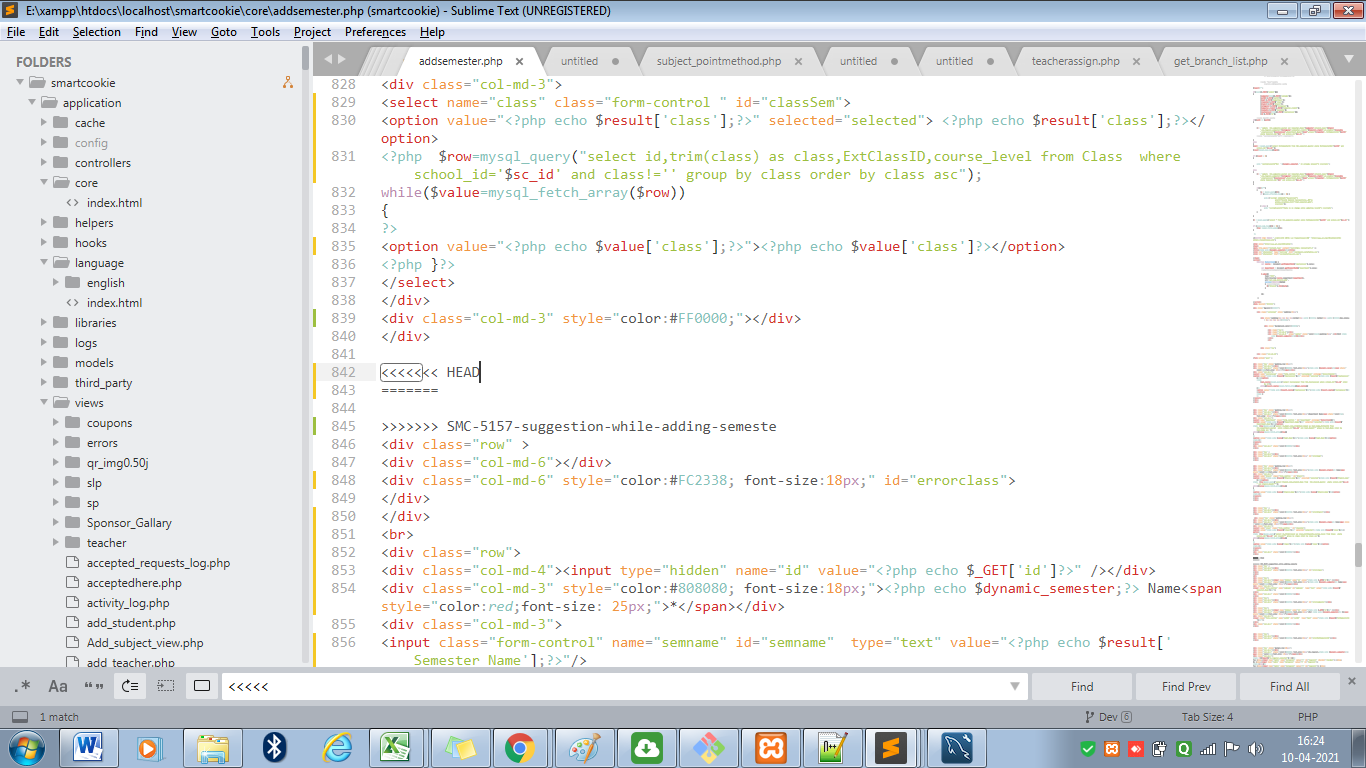
**If you don’t get conflicts follow below commands:**

1. On the new window opened as shown above, click the “INSERT” button from your keyboard. And type “SMC-5157(this will be your branch ID) merged”. Now press “ESC” button from your keyboard and then “:wq” and click on “Enter”. Now will get back to the git cmd.
2. git push origin Dev/dev
3. Now after 5 minutes test the functionality/issue that you have resolved on Dev environment
4. You will get conflicts if you remain on same window and you get an error of “CONFLICT” as shown below:



**If you get conflicts then follow below commands:**

1. Open the file which is conflicted
2. Now check for ‘<<<<’ and you will find the conflict.



1. As you can see in above picture “<<<<<HEAD” indicates starting of your conflict. “======” indicates the middle layer of the conflict and “>>>>>branch name” indicates the end of the conflicts. So you have to decide which code you want to keep, the code above the “=====” or the code below that.
2. Don’t forget to erase “<<<<<HEAD”, “======” and “>>>>>branch name” from the code as they are there for just indicating the start, end and mid of the conflicts. Then save the file and go to your git cmd.
3. git status
4. git add file name (copy paste file path which we get after git status)
5. git status
6. git commit -m "SMC-5242 Solved conflicts"
7. git push origin Dev/dev
8. Now after 5 minutes test the functionality/issue that you have resolved on Dev environment

* **Giving Pull Request to test/Merging code on test if conflicted :-**

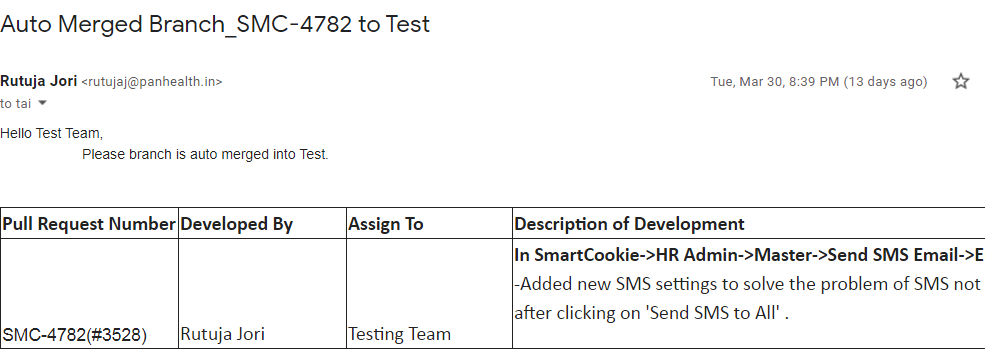
1. Go to Bitbucket
2. Select Repository
3. On the left side, select 4th option “Pull Requests”.
4. On right side, click on “Create Pull request”.
5. Now on left drop down search your branch ID (SMC-5242) and select the branch name and on right side select ‘test’ as shown below:



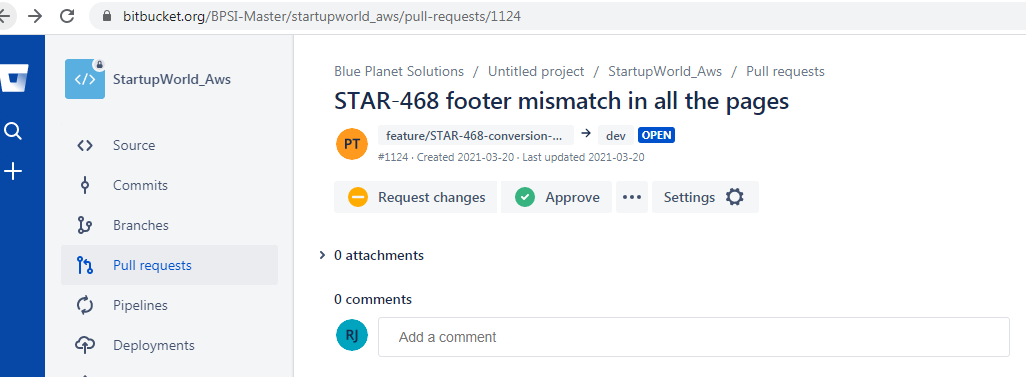
1. Now scroll down and click on “Create pull request”.
2. After creating pull request there will be 2 possibilities
3. You will get conflicts if following page appears with conflicts



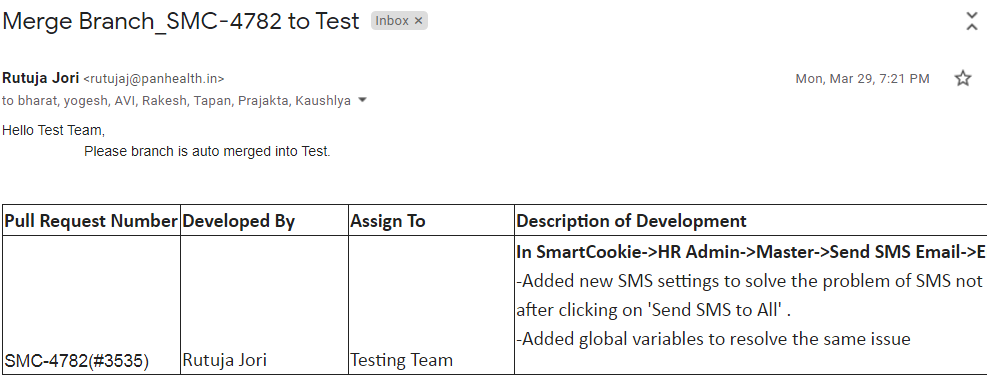
1. You can see in above image there is a message “This pull request can’t be merged. You will need to resolve conflicts to be able to merge”.
2. Now go to git cmd.
3. git checkout test
4. git pull origin test
5. git merge branch name
6. Now solve conflicts as mentioned above under ‘Dev’ section from the file which is conflicted.
7. git status
8. git add file name (copy paste file path which we get after git status)
9. git status
10. git commit -m "SMC-5242 Solved conflicts"
11. git push origin test
12. Now after 5 minutes test the functionality/issue that you have resolved on test environment
13. Mail the same about the changes done to testing team and Avi Sir and Rakesh Sir. Below is the format for the same:



1. You will not get conflicts if page appears without conflict message as shown below:



1. Mail the same about the changes done to testing team and Avi Sir and Rakesh Sir. Below is the format for the same :



1. Now go to JIRA, search the same issue which you have just resolved. Now its status is ‘In Progress’ change the same to ‘Done’ which will be the 5th button at top after the title of the issue. Now status will change to ‘Programming Done’.

* **Steps for solving conflicts on master :-**

1. Tester or the networking team will inform you that your branch is conflicted and you need to resolve the same.
2. Now you need to check who is available in the networking team and inform him that you have to solve conflicts for ‘branch name’ under ‘project name’ and you need access of their pc to do so.
3. Now inside the Project folder, right click and do ‘git bash here’ (If already done ignore this step).
4. git pull
5. Now it will ask for password. Networking team will provide the same.
6. git checkout branch name.
7. git pull
8. git checkout master
9. git pull
10. Enter the same password again
11. git merge branch name
12. Now solve the conflicts from the files as mentioned above.
13. git status
14. git add filename
15. git commit -m “SMC-5242 Conflicts solved by your name”.
16. git push
17. Enter password
18. And you are done.