$ history

1 cd d:/

2 cd WorkSpaceMSS

3 git init

4 git clone https://github.com/mag-dev/mss-web-ui-test

9 cd mss-web-ui-test/

10 git checkout -b test1 origin/test1

11 git rebase origin/master

12 git mergetool

14 git rebase --continue

15 git mergetool

43 git rebase --skip

44 git rebase --continue

45 git mergetool

46 git rebase --continue

47 git mergetool

48 git rebase --continue

49 git mergetool

50 git rebase --skip

51 git rebase --continue

52 git mergetool

53 git mergetool

54 git rebase --continue

55 git mergetool

56 gitk

57 git rebase --skip

58 git status

60 git fetch --all

61 git status

62 git push -u origin test1

63 git push -f origin test1

64 git status

65 git checkout master

66 git merge test1

67 gitk

68 git status

69 git checkout master

70 git status

71 git merge test1

73 history

Steps to use client’s code:

Username/Pwd: Amohanani/annu1982

1) $ pwd

/c/Users/amohanani

Note: This command gives current working directory

2) $ cd /d/WorkSpace

Note: This command will change the current working directory. Run above command once again it will give /d/WorkSpace

3) git clone https://github.com/mag-dev/mss-web-ui-test.git

……..Checking connectivity… done

Note: This command will check out client repository into your local machine on D:/WorkSpace location

4) Open Visual Studio. File – Open – Select the project.

5) Right click on References and add the reference of selenium web driver.

6) To change browser to run tests:

Make change in BrowserTestingSessionFactory.cs in MAG.WebTesting.Browsers

public IBrowserTestingSession CreateSession<TMarker>(SupportedBrowserType browserType = SupportedBrowserType.Firefox)

7) To change URL:

BaseBrowserFixture in MAG.WebTesting.Browsers in GetPath

8) Search any text in VS – Ctrl+Shift+F

9) To add any wrapper method of Selenium add into BaseBrowserInstance and declare it into IBrowserInstance of MAG.WebTesting.Browsers

**gitadmin$ history**

1 cd /d/WorkSpace

2 git clone https://github.com/mag-dev/mss-web-ui-test

3 git init

4 git checkout -b test1

5. Opened Project in Visual studio

6. in git bash 🡪

Cd /d/WorkSpace/mss-web-ui-test

git checkout -b test1 origin/test1

This will create a new branch test1 in local and map it to remote test1 branch

7. git branch

Master

\*test1

Git push –u origin test1 -> This will create a new branch in remote repo named test1

Git –avv -> Check with which yr branch is in sync right now

\*\*\*\*git rebase origin/master

\*\*\*\*git mergetool

**Command’s in Tester’s machine:**

cd /d/WorkSpace

git clone file:////INMDCD0563/mss-web-ui-test

git init

git checkout -b test3

Modified file

commit & Sync with comments from VS

Git bash -> git push --set-upstream origin test3

**GitAdmin’s machine:**

Git checkout test3

* This will change to branch test3
* Again back to test1 (git checkout test1)
* Git merge test3