

Sitecore-Applications PushToMaster User documentation

1. Requirement Details:

PushToMaster script is designed after a requirement particular to Sitecore team and here are the requirement details:

1. Sitecore has a module for Sitecore-applications along with CM and CD modules.
 2. All the code of this module is maintained in Sitecore-applications repository of GitHub.
 3. This repository has two branches. One branch is named as "Master," and the other branch is named as "dev-branch."
 4. All the development is done in "dev-branch."
 5. Hence all the code changes like fixing the bugs, code improvisations, code development happens in "dev-branch."
 6. "master" branch is maintained clean all the time.
 7. Deployment of code to servers is also done using "master" branch code.
 8. Hence the current version of code running on the server, which is available to users, is same as the code in the master branch.
 9. Once if all the changes in "dev-branch" are tested and decided to be ready for deployment, these changes have to be implemented in the master branch.
 10. This can be achieved by just using "Merge" option in Git.
 11. By checking out to "master" branch and merging the changes from "dev-branch" to master, all the changes in "dev-branch" gets reflected in "master," along with the commit history of "dev-branch."
 12. Here comes the primary requirement that, even though dev-branch have all commit history, only the history of changes should be reflected in "master" branch.
 13. That means, commit history of "master" branch should be clean; however, it has to be updated with the changes in "master-branch."
- For example, I am working in the master branch.

```
PS C:\WorkStation\GitLocalRepo\Azure-Auto-deploy> git status
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
```

- Now I am changing to (checking out to) dev-branch

```
PS C:\WorkStation\GitLocalRepo\Azure-Auto-deploy> git checkout dev-branch
Switched to branch 'dev-branch'
Your branch is up to date with 'origin/dev-branch'.
```

- I added a file to dev-branch and added some data to that file.

```
PS C:\WorkStation\GitLocalRepo\Azure-Auto-deploy> Vim abc.txt
```

- I saved the file and committed these changes to GitHub remote repository.
- Then I modified an existing file *index.html*, saved the file and committed to GitHub remote repository.
- Finally, I decided that I don't need *abc.txt* and deleted this file.
- Now, I committed these changes to remote GitHub repository.
- If I checkout to "master" branch and merge "dev-branch" to master, I will have all the history of adding *abc.txt*, modifying *index.html* and deleting *abc.txt*.
- But the actual change in "dev-branch" from "master" is just updating *index.html* file
- Just this change should be updated to "master" branch.

2. About the script:

1. To fulfill this requirement, firstly, the differences between "dev-branch" and "master" should be identified.
2. "Master" branch should be updated with all these differences.
3. Then these new changes of "master" branch should be committed to the remote GitHub repository.
4. As we are just updating the master branch with the changes and committing them, there will be only one commit history irrespective of the number of commits in "dev-branch."
5. This entire process is automated in a PowerShell script named "PushToMaster."

6. Techniques used in the script are

- “git diff” to find differences between branches and
- “git patch” to save the differences into a file to apply them to the “master” branch

3. Installing Script:

1. This script has to be placed in local git repository of the project for which you would like to apply this requirement.
2. That repository has to have both “master” as well as “dev-branch.”
3. You may also have other branches in that repository, but this script works only between “master” and “dev-branch.”
4. Include the name of this script to the .gitignore file.
5. By doing this, it will not be pushed to your remote git repository and this file gets ignored in all of your git activities.
6. Once all these steps got implemented, we are good to use the script.

4. Using the script:

1. Generally, for Sitecore-applications, we work from “dev-branch” as explained in [Requirement Details](#)
2. Once all the changes of “dev-branch” are tested and are ready to be committed to “master,” you can just run the script.
3. Here is the example that shows how the script works and how it saves history for “dev-branch” as well as “master.”

Step 1:

- This is the remote git repository of project Azure-Auto-deploy and we are in “dev-branch”
- Checking for git log that shows commit history of *the index.html* file in “dev-branch”
- Here are the commits made until May 7th, 2018.
- Commit5 is the most recent commit

History for [Azure-Auto-deploy](#) / index.html

Commits on May 7, 2018

Commit5

 anushaande committed 3 days ago

 0eeae33 

Commit2

 anushaande committed 3 days ago

 04e7101 

Commits on May 2, 2018

Committing changes from dev-branch

 anushaande committed 8 days ago

 ad183fe 

Step 2:

- Now let us check for the commit history of the same file in “master” branch.
- Checking for git log that shows commit history of index.html file in “master”
- Here are the commits made until May 7th, 2018 and today is May 10th, 2018
- “Committing changes on 05/07/2018 15:28:19” is the most recent commit.
- This looks different from “dev-branch”.

History for [Azure-Auto-deploy](#) / [index.html](#)

Commits on May 7, 2018

Committing changes on 05/07/2018 15:28:19

 anushaande committed 3 days ago



e12922a



Commits on May 2, 2018

Committing changes 2

 anushaande committed 8 days ago



8a4a527



committing changes1

 anushaande committed 8 days ago



e7aeeb9



Committing changes from dev-branch

 anushaande committed 8 days ago



ad183fe



Step 3:

- Switched to dev-branch to make some changes to *index.html* file and create some commit history.

```
PS C:\WorkStation\GitLocalRepo\Azure-Auto-deploy> git checkout dev-branch
Switched to branch 'dev-branch'
Your branch is up to date with 'origin/dev-branch'.
```

Step 4:

- Here is the commit history of “dev-branch” after changes are made and committed to the remote repository.
- We can see that there are few commits made today in this branch.
- commit6 to commit13 are the commits made on 10th May 2018.

anushaande / Azure-Auto-deploy

Watch

0

Star

0

Fork

0

<> Code

Issues 0

Pull requests 0

Projects 0

Wiki

Insights

Settings

Branch: dev-branch

Commits on May 10, 2018

commit13

anushaande committed 28 seconds ago

0118a35

<>

commit12

anushaande committed 3 hours ago

b7adee6

<>

commit11

anushaande committed 3 hours ago

58d6981

<>

commit10

anushaande committed 4 hours ago

b313d93

<>

commit9

anushaande committed 4 hours ago

377ece3

<>

commit8

anushaande committed 4 hours ago

5d55ef7

<>

commit7

anushaande committed 4 hours ago

bc3d121

<>

commit6

anushaande committed 4 hours ago

8dc533a

<>

Commits on May 7, 2018

Commit5

anushaande committed 3 days ago

0eeae33

<>

Commit2

anushaande committed 3 days ago

04e7101

<>

Step 5:

- Here is the commit history of “master”
- As it appears, I didn't make any changes in this branch on 10th may, 2018.

USF Health IS | 813.974.6288

anushaande / Azure-Auto-deploy

Watch 0 Star 0 Fork 0

Code Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

Branch: master

Commits on May 7, 2018

- Committing changes on 05/07/2018 15:28:19
anushaande committed 3 days ago [e12922a](#)
- commit1
anushaande committed 3 days ago [efa69a4](#)

Commits on May 4, 2018

- Committing changes on 05/04/2018 11:57:49
anushaande committed 6 days ago [7603cf4](#)
- Committing changes of PushToMaster
anushaande committed 6 days ago [f850994](#)
- Committing changes in pushtomaster to master
anushaande committed 6 days ago [5fbfe0a](#)
- Committing changes after deleting refs folder
anushaande committed 6 days ago [d0f1089](#)

Step 6:

- I would like to update the master branch with all those changes in “dev-branch” and maintain a clean commit history.
- For this, I have to run the script.
- I am in “dev-branch” now.
- It's always a best practice to run this script from “master”.
- Because script starts with checking to master branch.

```
PS C:\WorkStation\GitLocalRepo\Azure-Auto-deploy> git status
On branch dev-branch
Your branch is up to date with 'origin/dev-branch'.
```

```
nothing to commit, working tree clean
```

```
PS C:\WorkStation\GitLocalRepo\Azure-Auto-deploy> .\PushToMaster.ps1
```

```
-----Adding changes in dev-branch to master branch-----
-----Checking to master-----
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
-----Applying changes in dev-branch to master-----
Checking patch PushToMaster.ps1...
Checking patch index.html...
Applied patch PushToMaster.ps1 cleanly.
Applied patch index.html cleanly.
-----Committing the changes in master branch-----
[master b8b1a70] Committing changes on 05/10/2018 11:39:51
 2 files changed, 25 insertions(+), 41 deletions(-)
 rewrite PushToMaster.ps1 (72%)
-----Pushing changes to master branch-----
Username for 'https://github.com':
```

Step 7:

- Script gets paused at git push until we give credentials of remote git repository to push these changes to the master branch.
- The commit message is generated automatically with date and time of script execution.
- I changed two files, hence changes are made to those two files in master branch one after the other.

```

PS C:\WorkStation\GitLocalRepo\Azure-Auto-deploy> .\PushToMaster.ps1
-----Adding changes in dev-branch to master branch-----
-----Checking to master-----
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
-----Applying changes in dev-branch to master-----
Checking patch PushToMaster.ps1...
Checking patch index.html...
Applied patch PushToMaster.ps1 cleanly.
Applied patch index.html cleanly.
-----Committing the changes in master branch-----
[master 1979de1] Committing changes on 05/10/2018 14:24:58
 2 files changed, 25 insertions(+), 41 deletions(-)
 rewrite PushToMaster.ps1 (71%)
-----Pushing changes to master branch-----
Username for 'https://github.com':

```

Step 8:

- Here is the rest of the log after providing credentials to push the changes to remote repository.
- We can always ignore “fatal” that is highlighted.
- The remaining information is about the data that is getting pushed to the remote repository.
- Git status at the end shows that you are on the master branch and it's up to date with the remote repository.

```

-----Pushing changes to master branch-----
Username for 'https://github.com': anushaande
Password for 'https://anushaande@github.com':
Counting objects: 4, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 775 bytes | 258.00 KiB/s, done.
Total 4 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/anushaande/Azure-Auto-deploy.git
 e12922a..1979de1 master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
(fatal: Resolve operation not in progress, we are not resuming.)
-----Git status-----
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
PS C:\WorkStation\GitLocalRepo\Azure-Auto-deploy>

```

Step 9:

- This is how the script output looks like a whole

```

PS C:\WorkStation\GitLocalRepo\Azure-Auto-deploy> .\PushToMaster.ps1
-----Adding changes in dev-branch to master branch-----
-----Checking to master-----
Switched to branch 'master'
Your branch is up to date with 'origin/master'.
-----Applying changes in dev-branch to master-----
Checking patch PushToMaster.ps1...
Checking patch index.html...
Applied patch PushToMaster.ps1 cleanly.
Applied patch index.html cleanly.
-----Committing the changes in master branch-----
[master 1979de1] Committing changes on 05/10/2018 14:24:58
 2 files changed, 25 insertions(+), 41 deletions(-)
 rewrite PushToMaster.ps1 (71%)
-----Pushing changes to master branch-----
Username for 'https://github.com': anushaande
Password for 'https://anushaande@github.com':
Counting objects: 4, done.
Delta compression using up to 8 threads.
Compressing objects: 100% (4/4), done.
Writing objects: 100% (4/4), 775 bytes | 258.00 KiB/s, done.
Total 4 (delta 3), reused 0 (delta 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/anushaande/Azure-Auto-deploy.git
   e12922a..1979de1  master -> master
Branch 'master' set up to track remote branch 'master' from 'origin'.
fatal: Resolve operation not in progress, we are not resuming.
-----Git status-----
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
PS C:\WorkStation\GitLocalRepo\Azure-Auto-deploy>

```

Step 10:

- This is how commit history of the "master" branch after executing the script and applying changes.

anushaande / Azure-Auto-deploy
Watch 0
Star 0
Fork 0

<> Code
Issues 0
Pull requests 0
Projects 0
Wiki
Insights
Settings

History for Azure-Auto-deploy / index.html

Commits on May 10, 2018

Committing changes on 05/10/2018 14:24:58
anushaande committed 4 days ago
1979de1
<>

Commits on May 7, 2018

Committing changes on 05/07/2018 15:28:19
anushaande committed 7 days ago
e12922a
<>

Commits on May 2, 2018

Committing changes 2
anushaande committed 12 days ago
8a4a527
<>

committing changes1
anushaande committed 12 days ago
e7aeeb9
<>

Committing changes from dev-branch
anushaande committed 12 days ago
ad183fe
<>

Step 11:

- I added another line in the script to checkout to dev-branch once all the changes got pushed to the remote.
- This way, you will be in the same branch (dev-branch).

```
Branch 'master' set up to track remote branch 'master' from 'origin'.
fatal: Resolve operation not in progress, we are not resuming.
-----Git status-----
On branch master
Your branch is up to date with 'origin/master'.

nothing to commit, working tree clean
-----Going back to dev-branch-----
Switched to branch 'dev-branch'
Your branch is up to date with 'origin/dev-branch'.
PS C:\WorkStation\GitLocalRepo\Sitecore-Applications-Anusha> git status
On branch dev-branch
Your branch is up to date with 'origin/dev-branch'.

nothing to commit, working tree clean
```