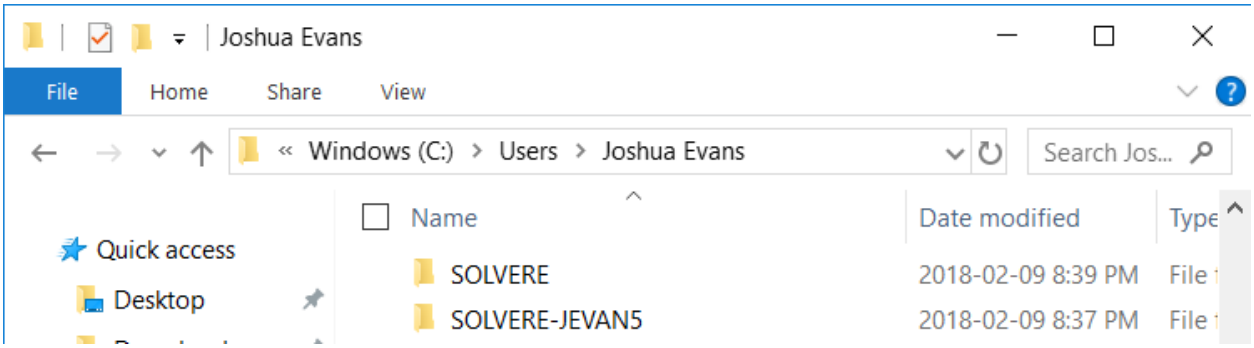
# Overview

To push to your branch, you must first checkout your branch. *git checkout *branchname** will delete all the files in your current local directory (whichever directory you make the GitBash command in), then pull all the files from the remote branch into your newly cleared repository. If you were in your local directory for SOLVERE, and executed a git checkout of your branch, added your new files to it, push your changes to your branch, then execute git checkout master to get the stable release back on your computer, you would only be able to pull what is on GitHub of our system.

This is a problem because our GitHub repository does not have all the files for *front-end/node_modules*. So, you would have to execute *ember new front-end* to generate the front-end/node_modules again. However, you were able to pull the front-end folder from GitHub, so the CLI will not allow this. Instead, you must delete the front-end folder, then execute *ember new front-end*, then pull *front-end* from GitHub again. You'd have to essentially walk through most of the SettingUpMEEN document again. Instead, here's a simpler way to push your changes to GitHub for the version controller to track.

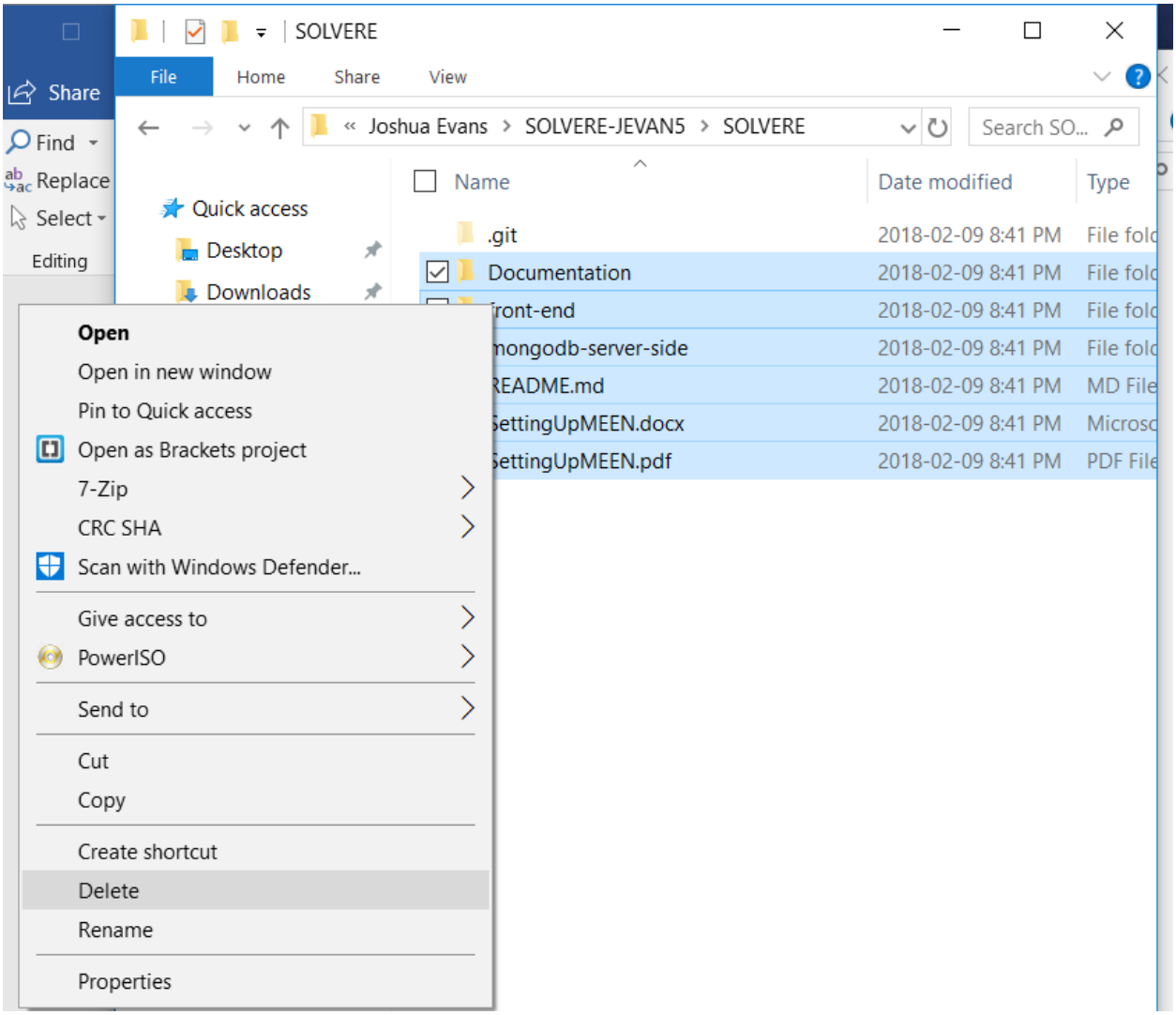
# Create A Folder For Your Branch

1. For simplicity's sake, create a folder next to your SOLVERE folder (I name mine *SOLVERE-JEVAN5*)

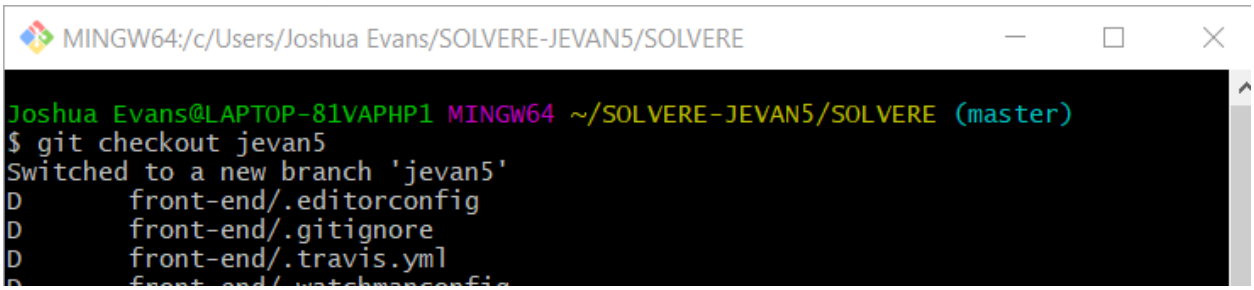



2. GitBash inside your new folder, and clone our repository 'git clone git@github.com:UWO-ECE-Software-Engineering/SOLVERE.git'

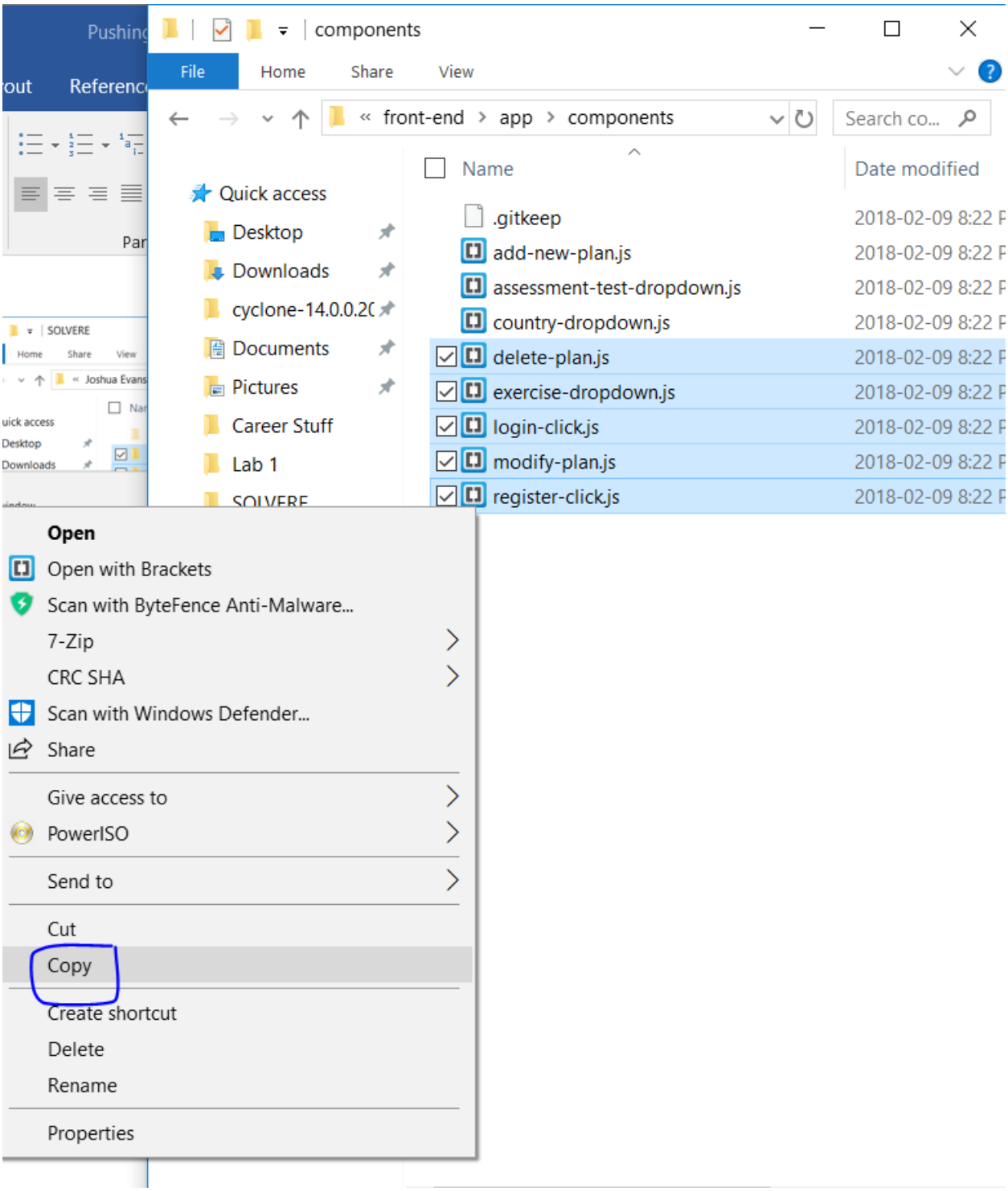
3. Inside *your-new-folder*/SOLVERE, delete everything except the folder '.git'

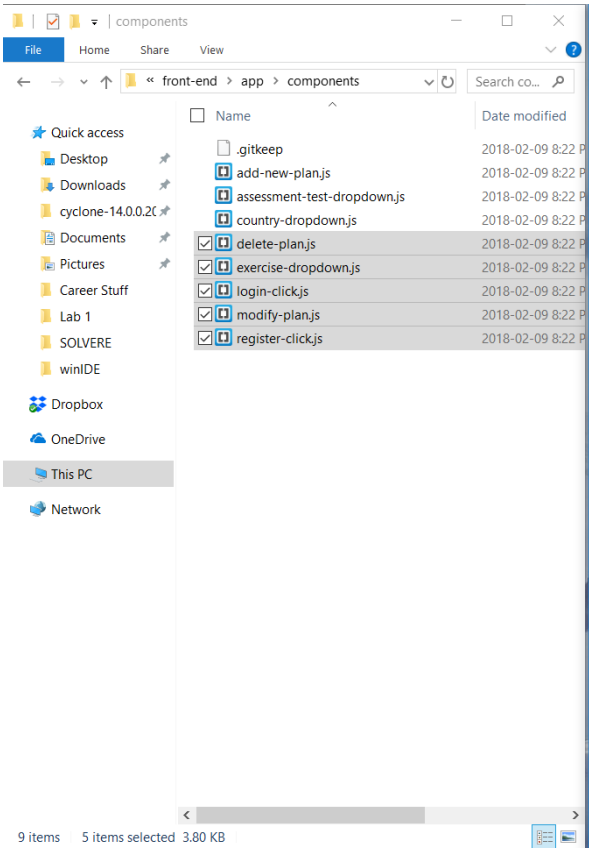


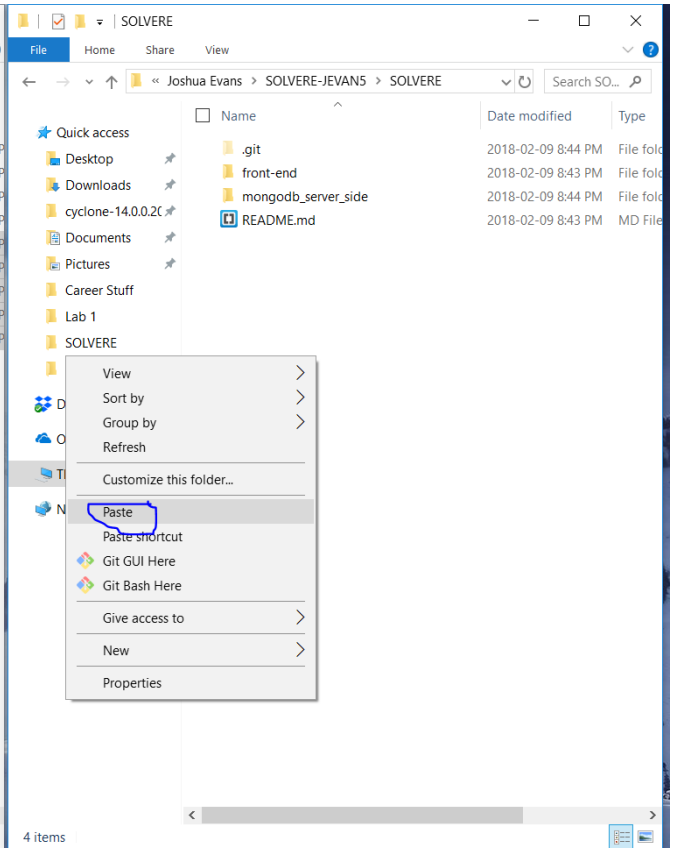

4. GitBash in here, and execute 'git checkout **your-branch-name**

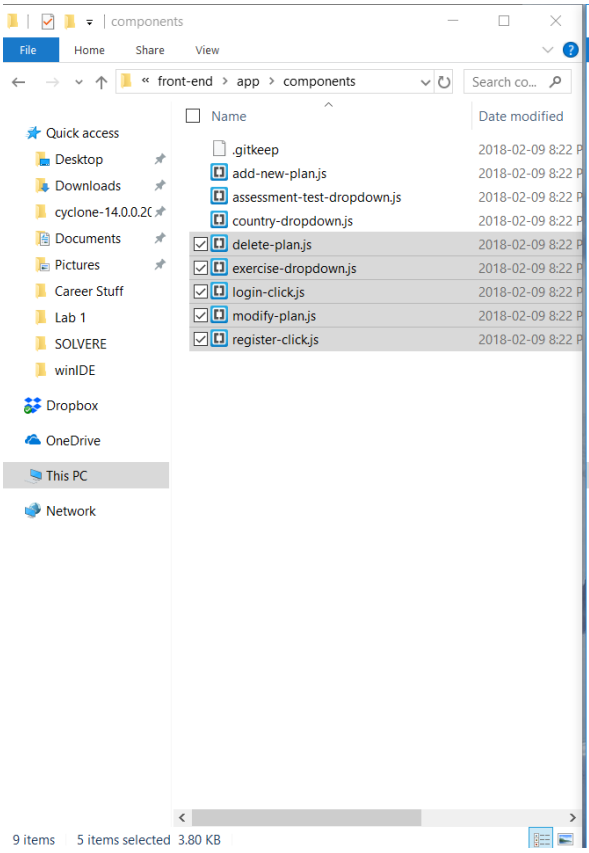


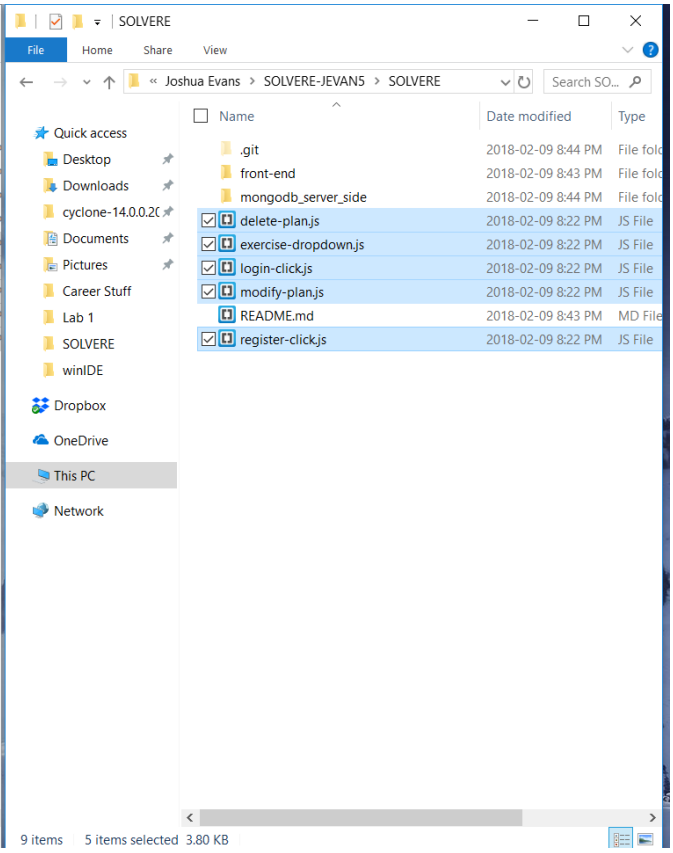

5. Now, whenever you make changes within your original SOLVERE folder (not the *your-new-folder/SOLVERE*), simply copy the folders and/or files over to this NEW folder (*your-new-folder/SOLVERE*)


components
File
Home
Share
View
front-end › app › components
Search co...
Quick access
Desktop
Downloads
Documents
Pictures
Career Stuff
Lab 1
Name
Date modified
.gitkeep
add-new-plan.js
assessment-test-dropdown.js
country-dropdown.js
delete-plan.js
exercise-dropdown.js
login-click.js
modify-plan.js
register-click.js
2018-02-09 8:22 P
Open
Open with Brackets
Scan with ByteFence Anti-Malware...
7-Zip
CRC SHA
Scan with Windows Defender...
Share
Give access to
PowerISO
Send to
Cut
Copy
Create shortcut
Delete
Rename
Properties

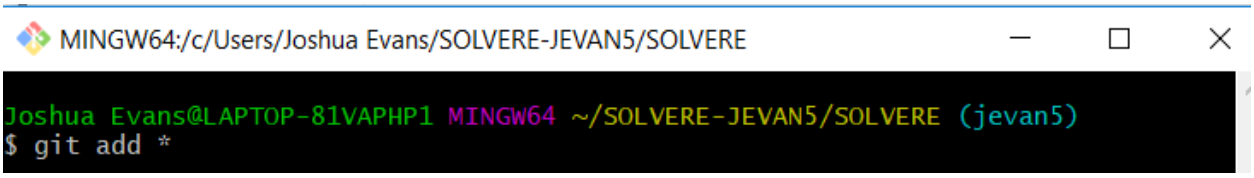

components
File
Home
Share
View
front-end › app › components
Search co...
Quick access
Desktop
Downloads
cyclone-14.0.0.2(
Documents
Pictures
Career Stuff
Lab 1
SOLVERE
winIDE
Dropbox
OneDrive
This PC
Network
Name
Date modified
.gitkeep 2018-02-09 8:22 P
add-new-plan.js 2018-02-09 8:22 P
assessment-test-dropdown.js 2018-02-09 8:22 P
country-dropdown.js 2018-02-09 8:22 P
delete-plan.js 2018-02-09 8:22 P
exercise-dropdown.js 2018-02-09 8:22 P
login-click.js 2018-02-09 8:22 P
modify-plan.js 2018-02-09 8:22 P
register-click.js 2018-02-09 8:22 P
9 items
5 items selected 3.80 KB


SOLVERE
File
Home
Share
View
Joshua Evans › SOLVERE-JEVAN5 › SOLVERE
Search SO...
Quick access
Desktop
Downloads
cyclone-14.0.0.2(
Documents
Pictures
Career Stuff
Lab 1
SOLVERE
Name
Date modified
Type
.git 2018-02-09 8:44 PM File fold
front-end 2018-02-09 8:43 PM File fold
mongodb_server_side 2018-02-09 8:44 PM File fold
README.md 2018-02-09 8:43 PM MD File
View
Sort by
Group by
Refresh
Customize this folder...
Paste
Paste shortcut
Git GUI Here
Git Bash Here
Give access to
New
Properties
4 items

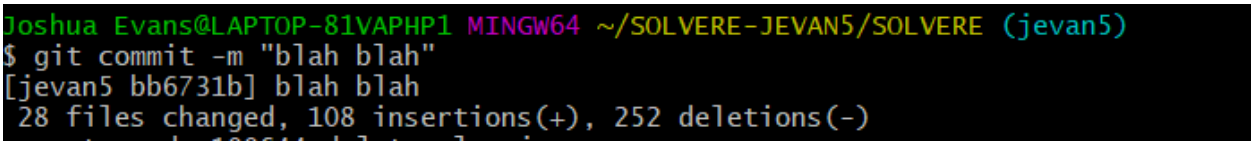

components
File
Home
Share
View
front-end › app › components
Search co...
Quick access
Desktop
Downloads
cyclone-14.0.0.2(
Documents
Pictures
Career Stuff
Lab 1
SOLVERE
winIDE
Dropbox
OneDrive
This PC
Network
Name
Date modified
.gitkeep 2018-02-09 8:22 P
add-new-plan.js 2018-02-09 8:22 P
assessment-test-dropdown.js 2018-02-09 8:22 P
country-dropdown.js 2018-02-09 8:22 P
delete-plan.js 2018-02-09 8:22 P
exercise-dropdown.js 2018-02-09 8:22 P
login-click.js 2018-02-09 8:22 P
modify-plan.js 2018-02-09 8:22 P
register-click.js 2018-02-09 8:22 P
9 items
5 items selected 3.80 KB


SOLVERE
File
Home
Share
View
Joshua Evans › SOLVERE-JEVAN5 › SOLVERE
Search SO...
Quick access
Desktop
Downloads
cyclone-14.0.0.2(
Documents
Pictures
Career Stuff
Lab 1
SOLVERE
winIDE
Dropbox
OneDrive
This PC
Network
Name
Date modified
Type
.git 2018-02-09 8:44 PM File fold
front-end 2018-02-09 8:43 PM File fold
mongodb_server_side 2018-02-09 8:44 PM File fold
delete-plan.js 2018-02-09 8:22 PM JS File
exercise-dropdown.js 2018-02-09 8:22 PM JS File
login-click.js 2018-02-09 8:22 PM JS File
modify-plan.js 2018-02-09 8:22 PM JS File
README.md 2018-02-09 8:43 PM MD File
register-click.js 2018-02-09 8:22 PM JS File
9 items
5 items selected 3.80 KB

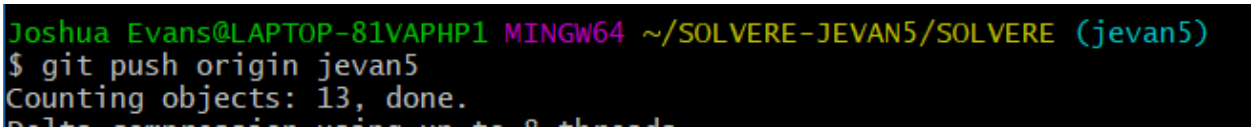
6. Navigate to *your-new-folder/SOLVERE* in GitBash, and execute 'git add *'




7. Execute git commit -m "*some message*"




8. Execute 'git push origin *your-branch-name*'




## Notes

All your work will still happen in your original SOLVERE folder, but you are simply going to push your changes from your own branch. After every change to the stable release on GitHub (these will not be very frequent, probably every 2 weeks), you will have to follow the SettingUpMEENStack document, starting from the GitHub section. At the end of every 2 weeks, you will have to store your changes in your local branch, then push your work.

For the ease of merging everyone's work, it would really be beneficial if you did not simply add the files that you changed. Instead, create the folder and file structure for any files that you are modifying.

For example, if I change the file front-end/app/components/add-new-plan.js, I wouldn't simply have my branch contain the folder add-new-plan.js. Instead, within my local branch I would create the folder front-end, then within front-end create the folder app, then within the folder app create the folder components, then within the folder components copy my add-new-plan.js file.

> **TO BE CLEAR:** You do not add ALL the files that happen to be in the same folder of the file you modified (In the example within the paragraph above, I would not add all the files within front-end/app/components/ into my branch)

The reason for this is that most of the files you are going to be changing are .hbs and .js files. .hbs and .js files can belong in many different folders in our project. The version controller would need to communicate with every team member about every single file that they simply through in their branch.