FTC New Platform Workshop

presented

By

FTC TEAM #8565

Setup GitHub for Team Development

Brandon Wang

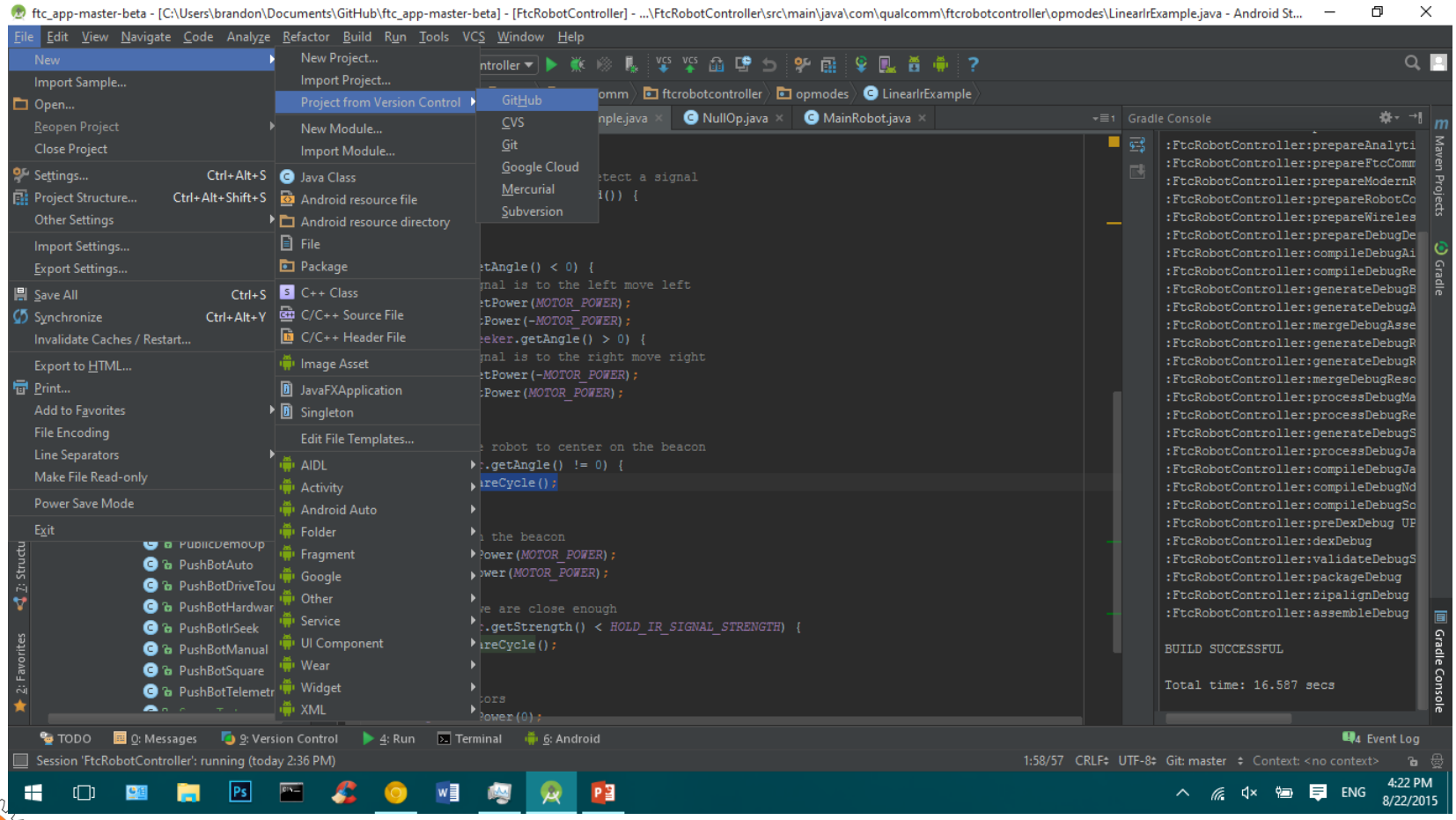
Step 1: Making and Sharing

Creating GitHub accounts

All of the programmers will need one!

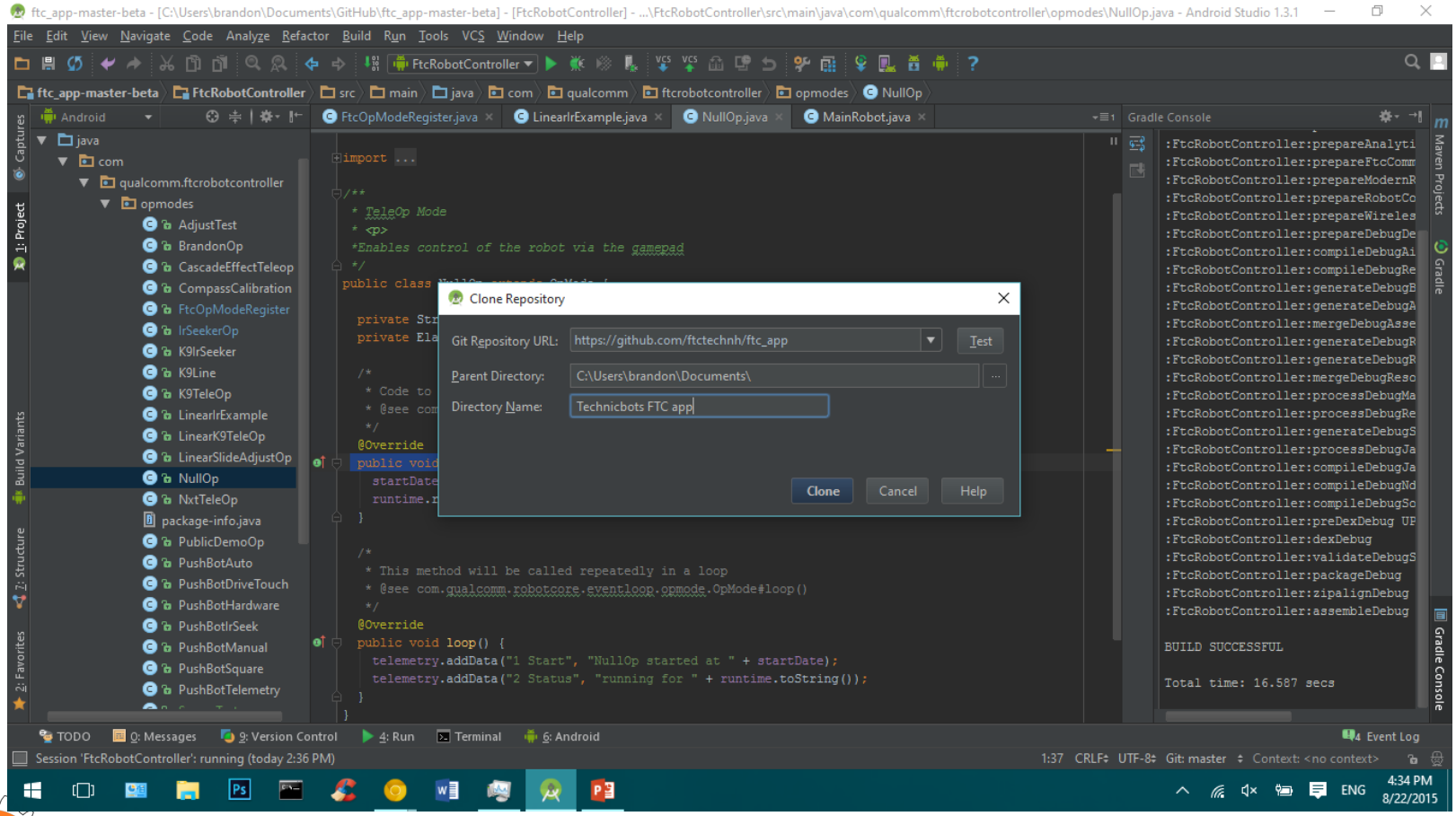
Importing project from remote Github repository

Go to File > New > Project from Version Control > Github.



FTC SDK Repository

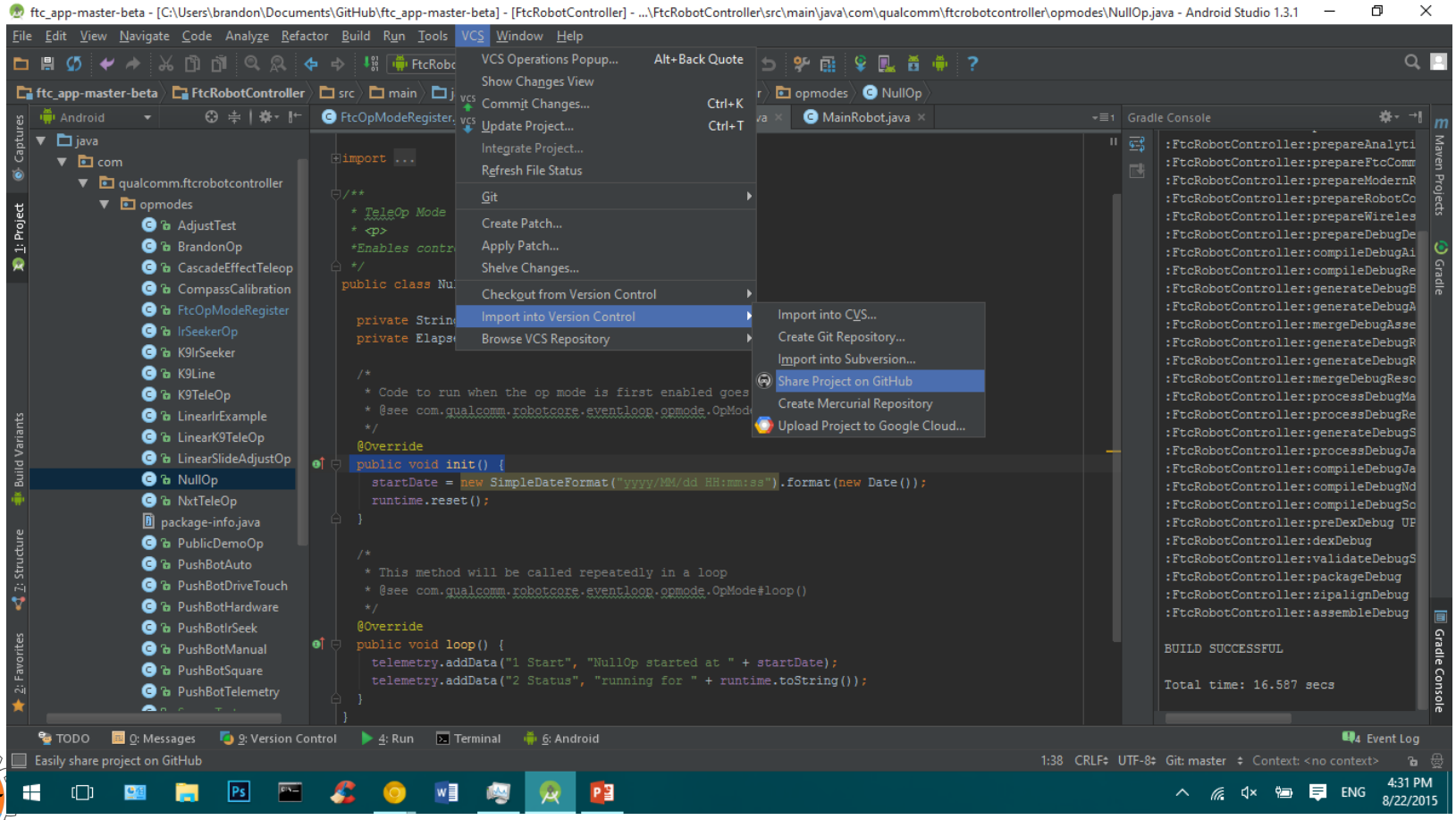
The FTC repository is located at https://github.com/ftctechnh/ftc_app



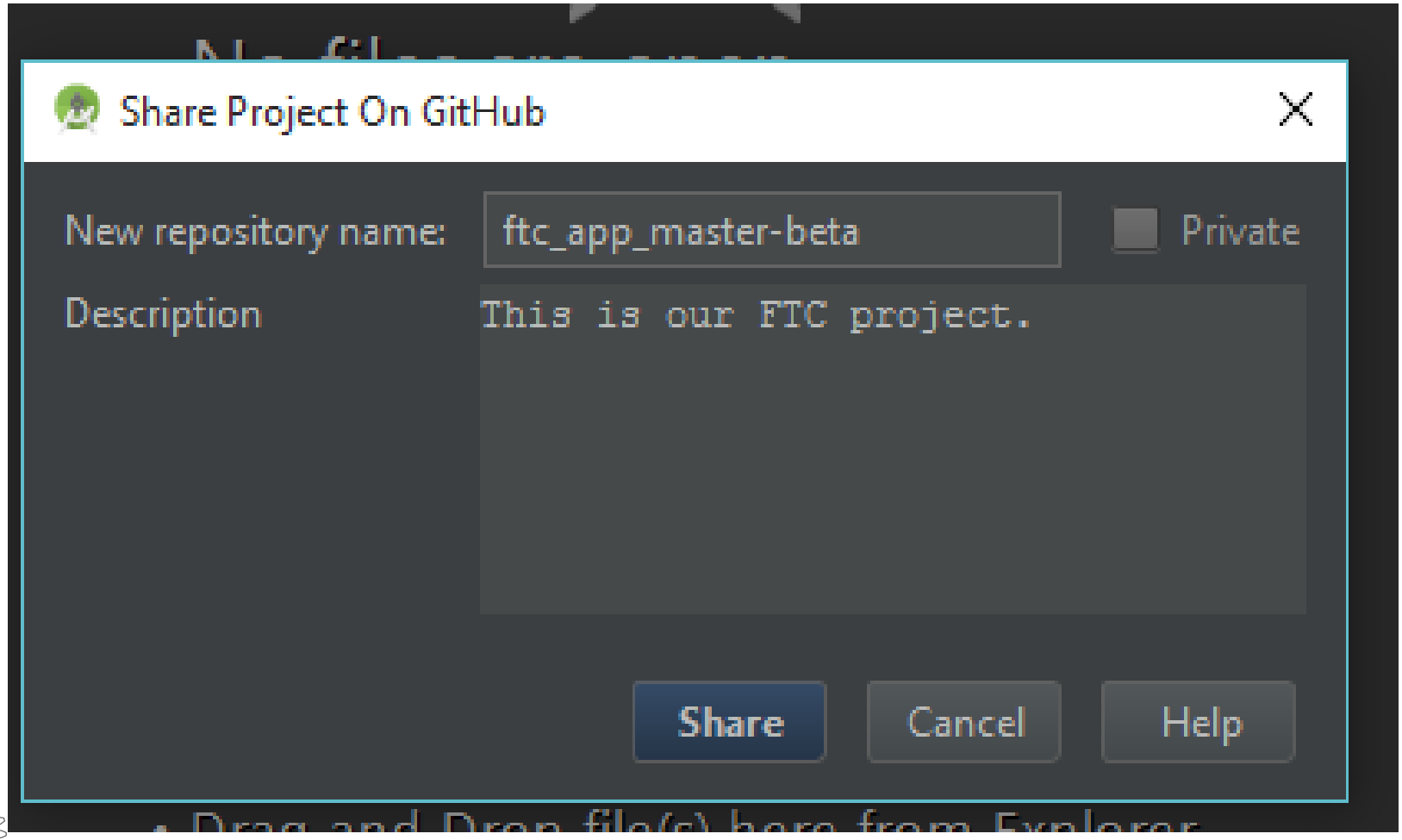
Uploading your copy of the FTC project to GitHub

Uploading your copy of the FTC project to GitHub

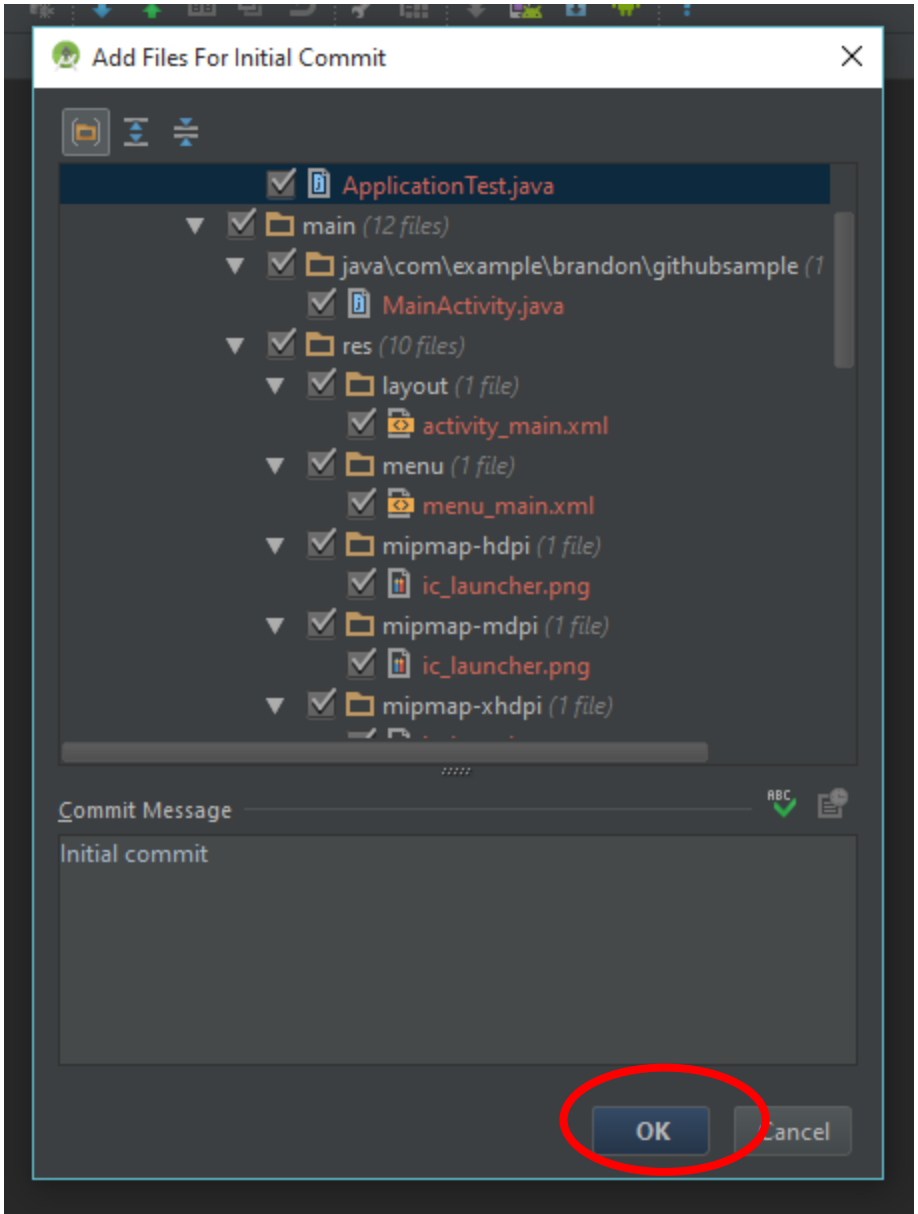
VCS > Import into Version Control > Share Project on Github.



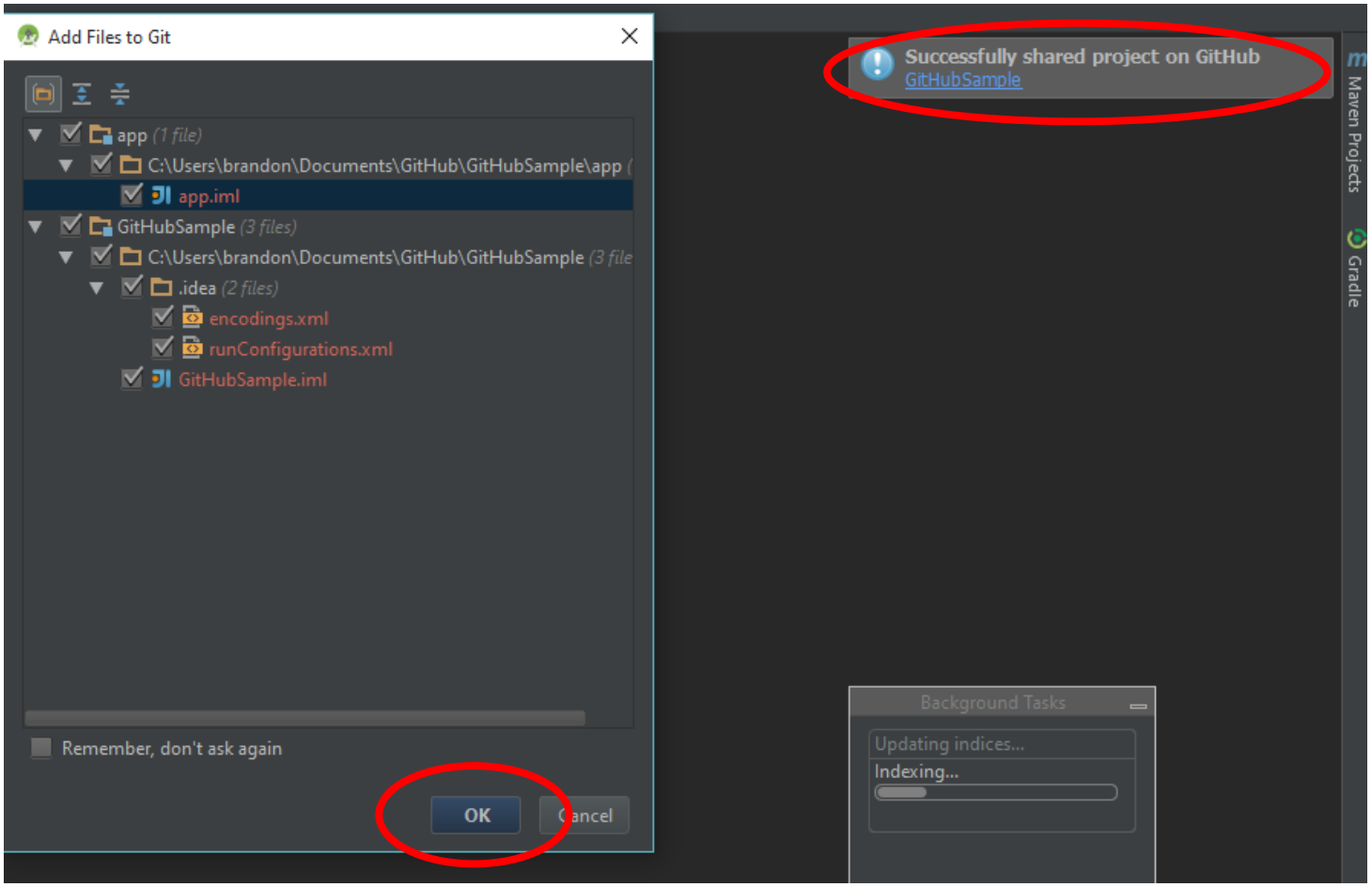
Uploading your copy of the FTC project to GitHub



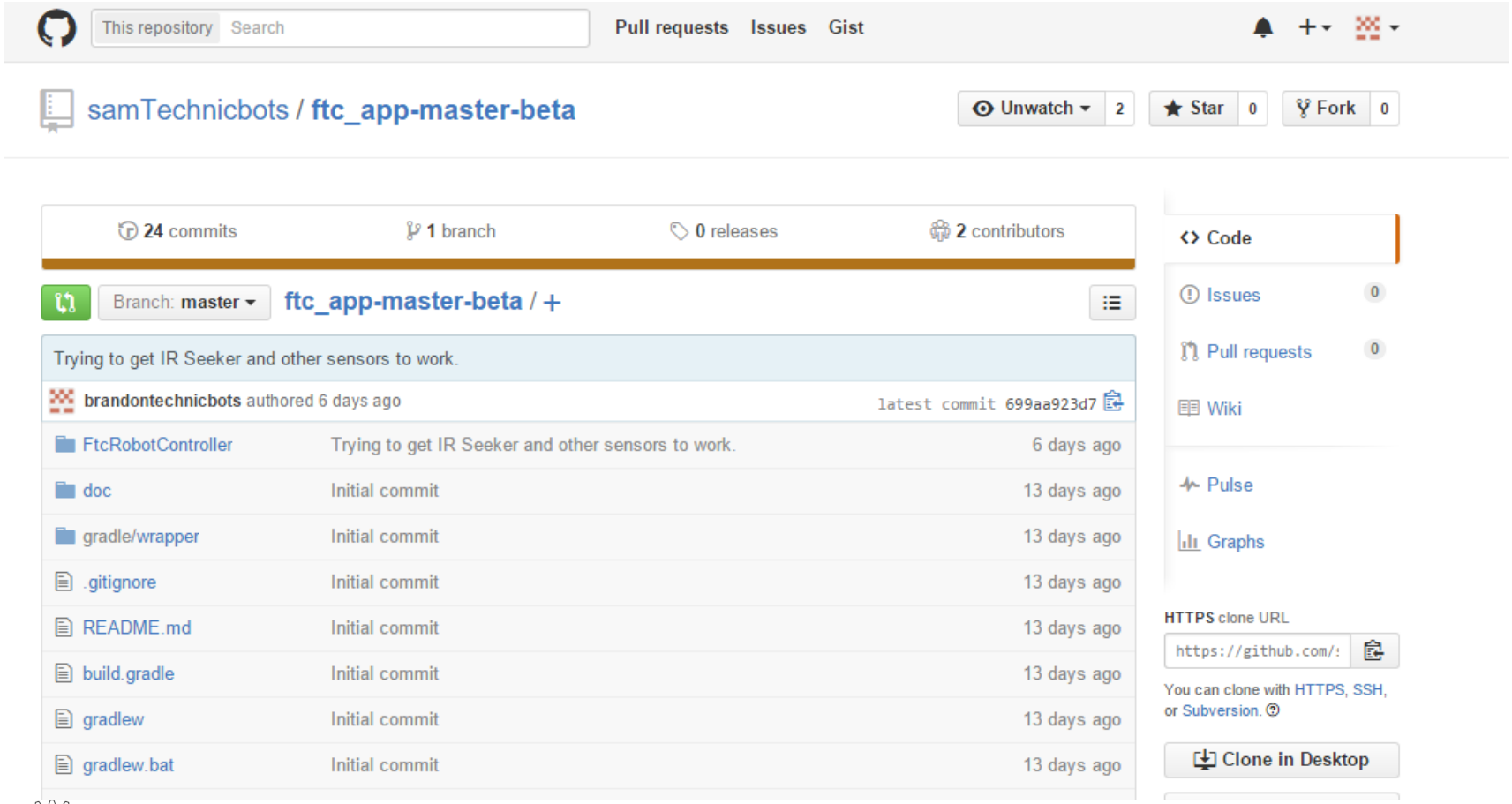
Uploading your copy of the FTC project to GitHub



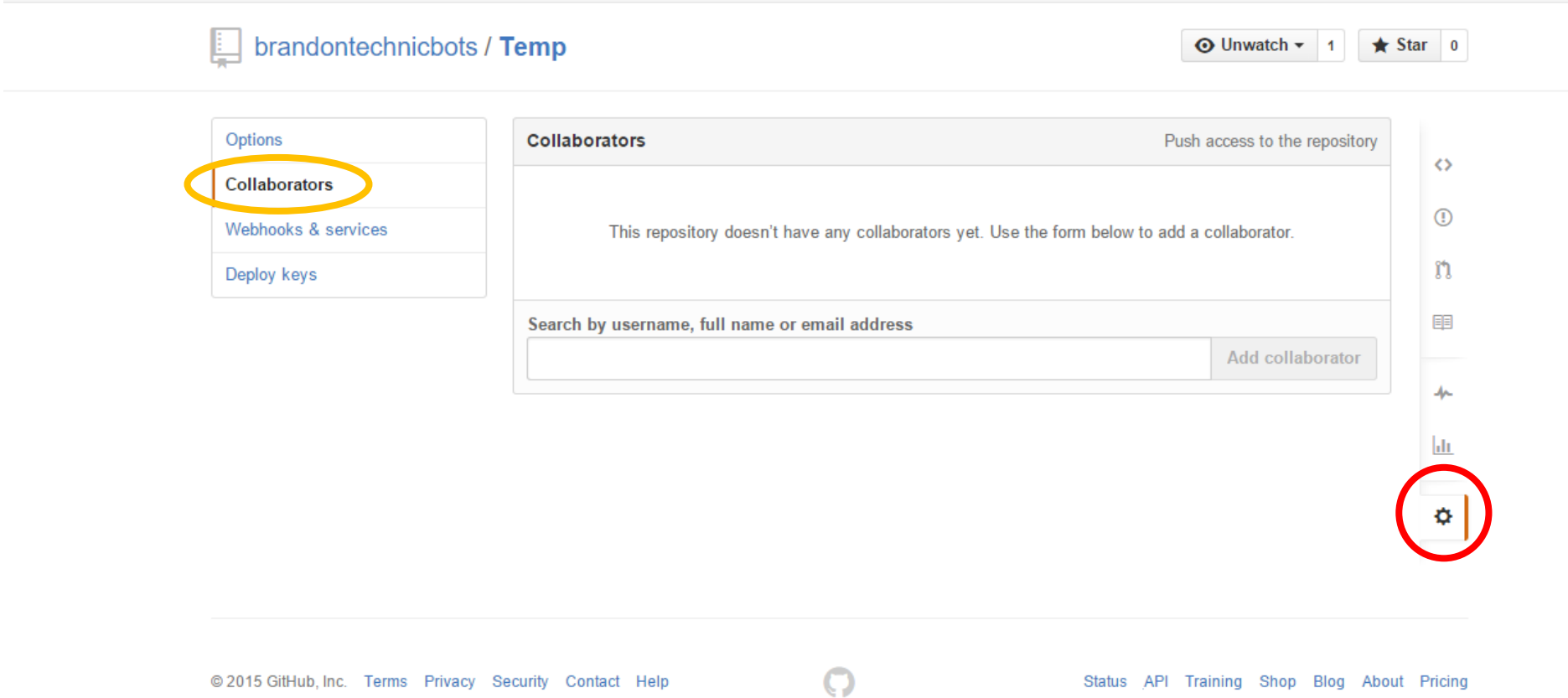
Uploading your copy of the FTC project to GitHub



Your copy of the FTC project on GitHub

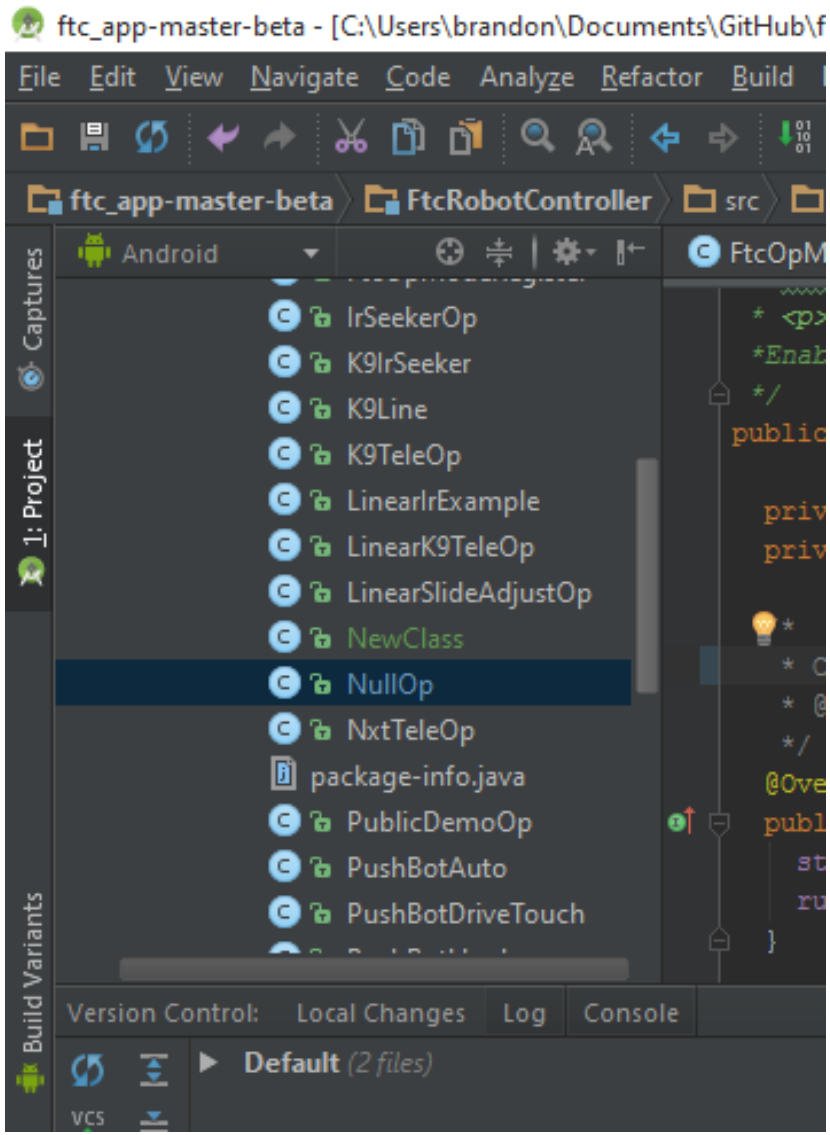


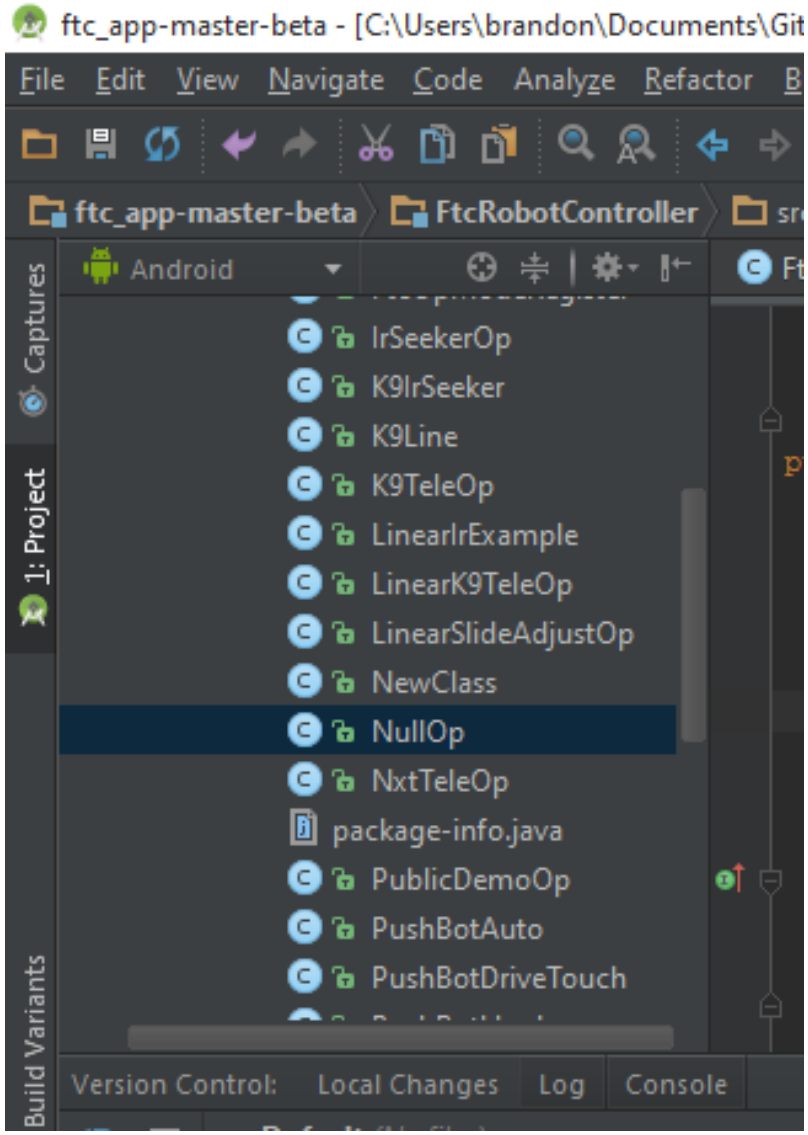
Adding Team Members (On the GitHub site)



Step 2: The Workflow

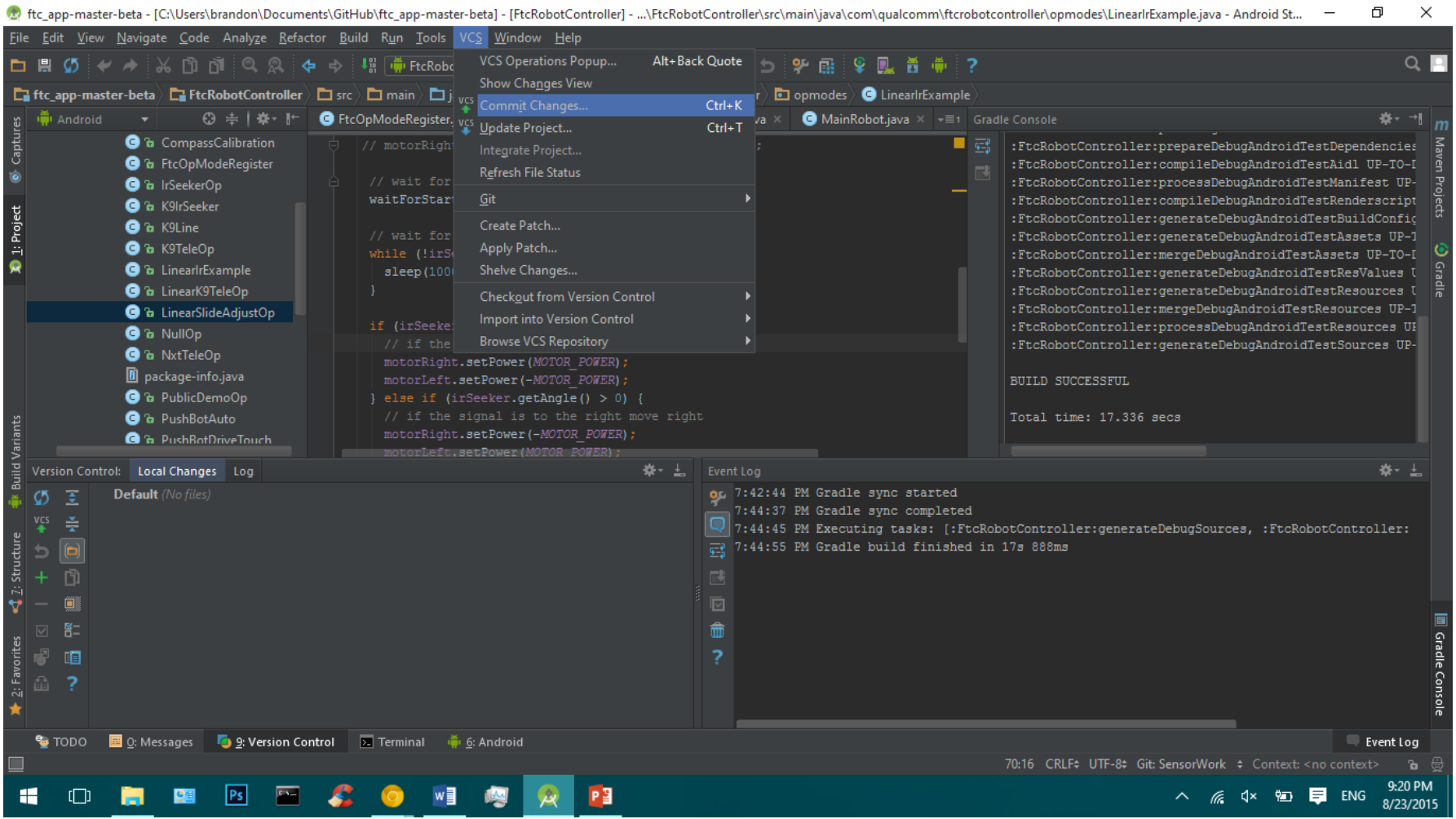
Color Coding





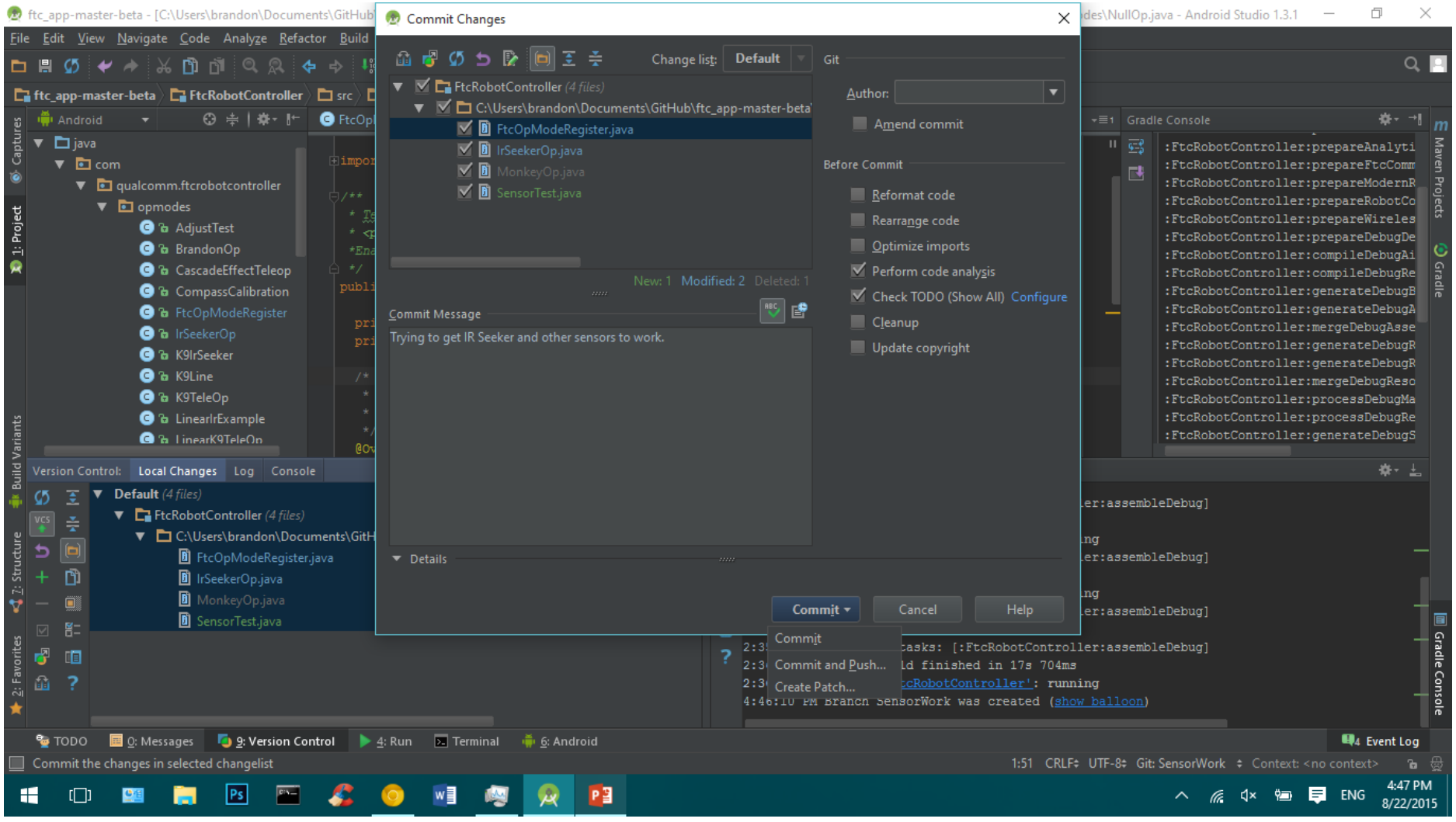
Making Commits

- VCS > Commit Changes (Or Ctrl+K).



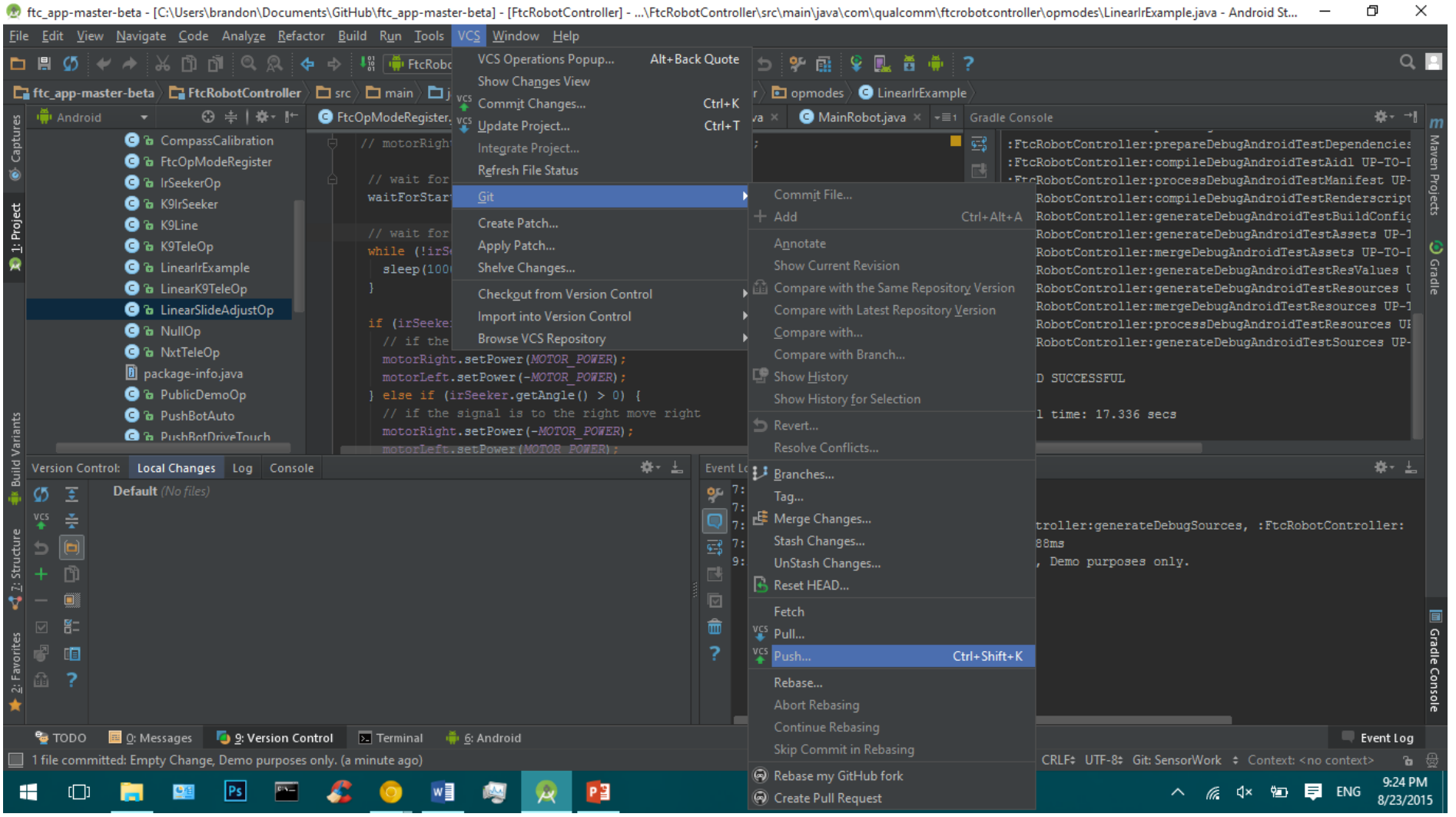
Making Commits

- Remember to write a detailed description to help identify changes later.



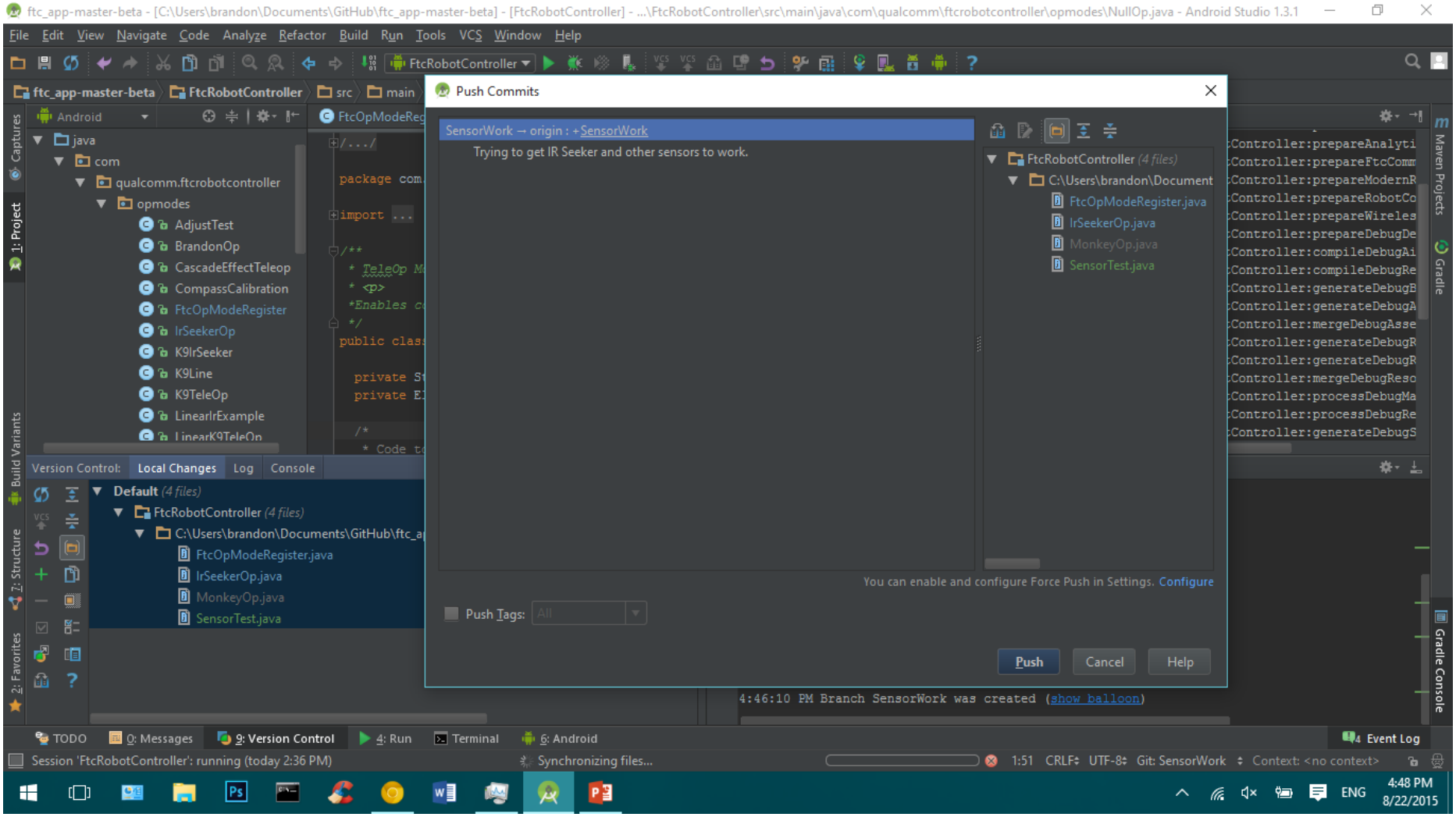
Pushing

VCS > Git > Push (Or Ctrl+Shift+K)

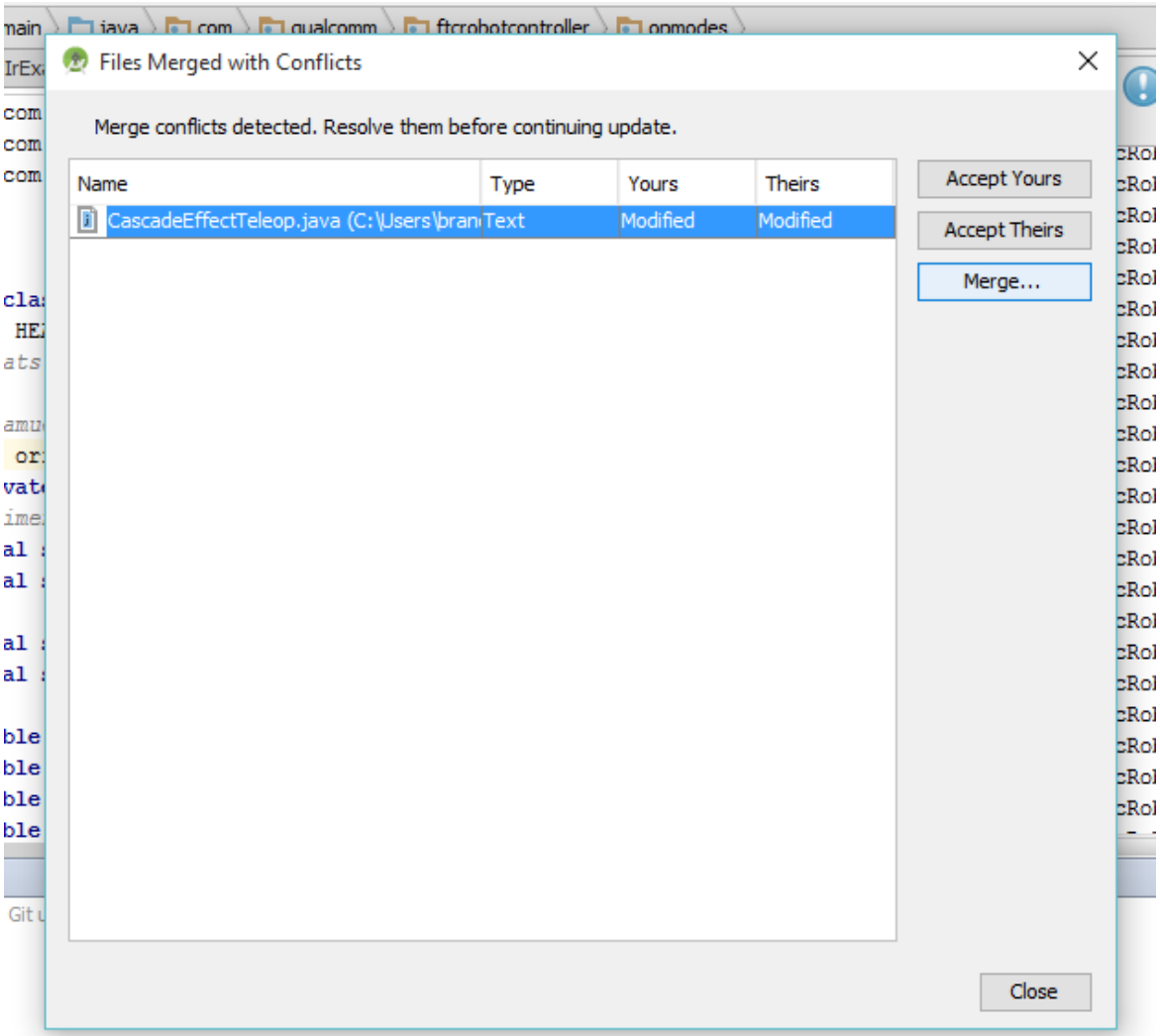


Pushing

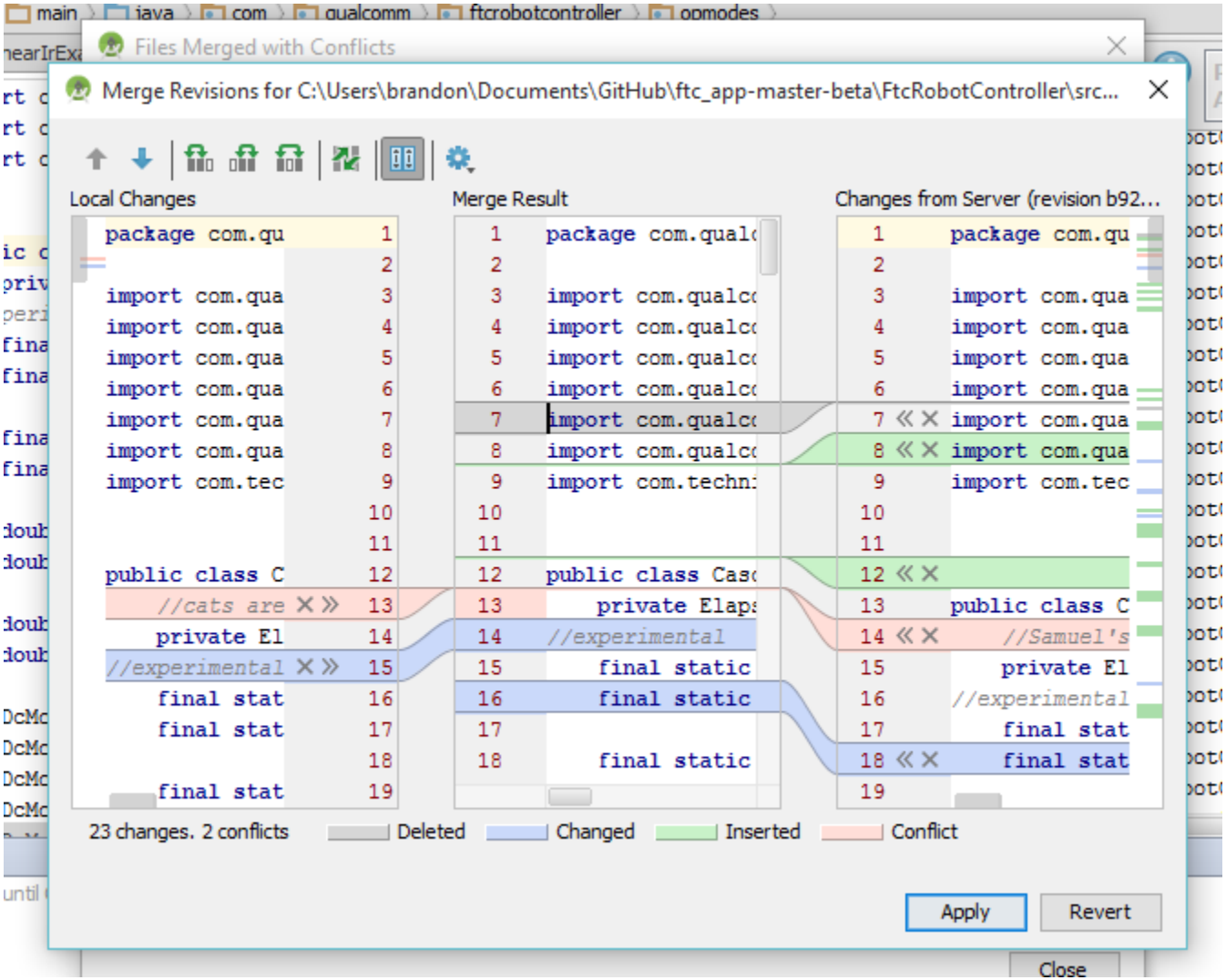
- Here you can choose what commits you want to push.



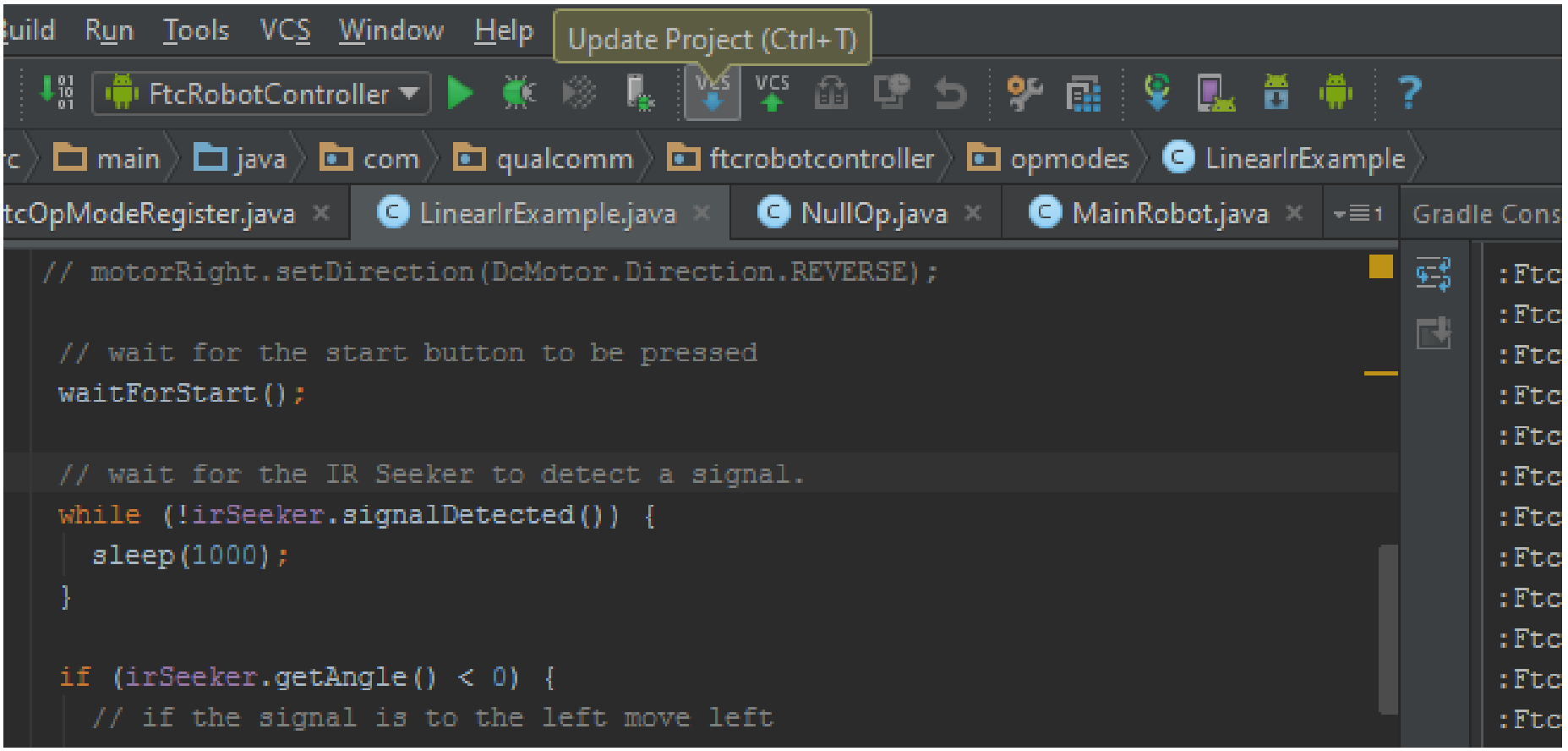
Pushing (Resolving Conflicts)



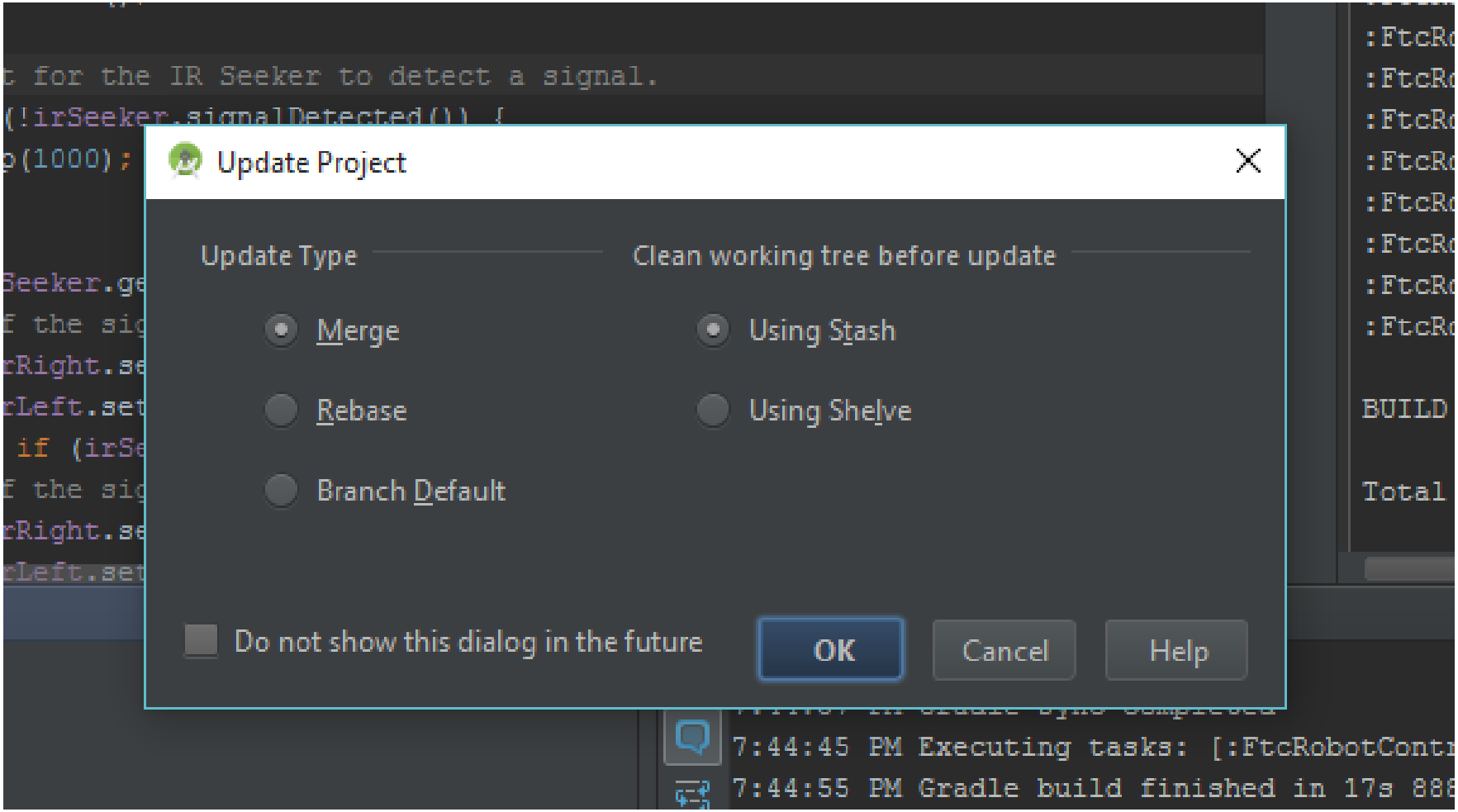
Pushing (Resolving Conflicts)



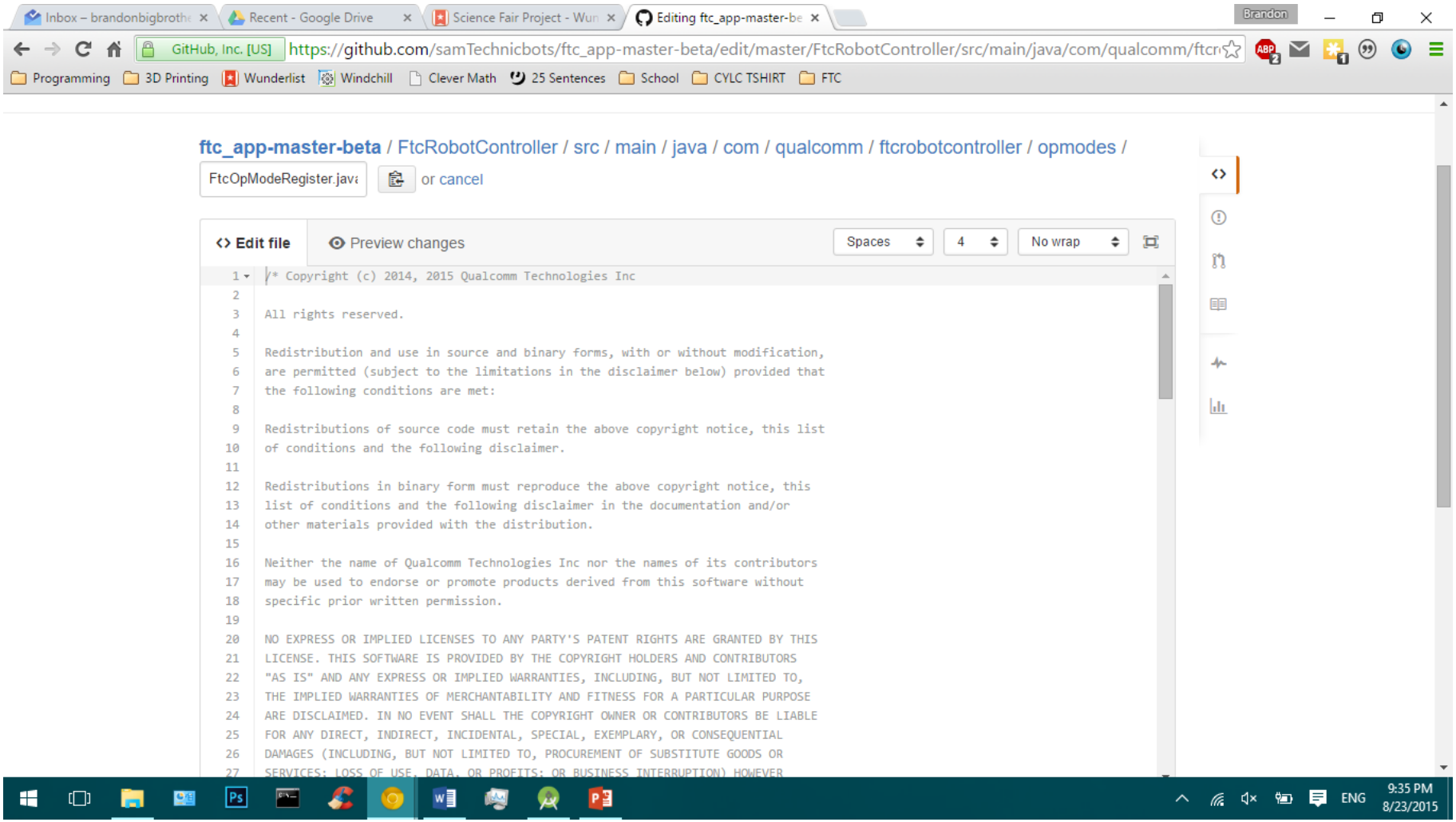
Pulling


tHub\ftc_app-master-beta] - [FtcRobotController] - ...\FtcRobotController\src\main\java\com\qualcomm\ftcrobotcontrolle
uild Run Tools VCS Window Help
Update Project (Ctrl+T)
FtcRobotController
main java com qualcomm ftcrobotcontroller opmodes LinearIrExample
tcOpModeRegister.java
LinearIrExample.java
NullOp.java
MainRobot.java
Gradle Cons
// motorRight.setDirection(DcMotor.Direction.REVERSE);
// wait for the start button to be pressed
waitForStart();
// wait for the IR Seeker to detect a signal.
while (!irSeeker.signalDetected()) {
sleep(1000);
}
if (irSeeker.getAngle() < 0) {
// if the signal is to the left move left

Pulling


Update Project
Update Type
Merge
Rebase
Branch Default
Clean working tree before update
Using Stash
Using Shelve
Do not show this dialog in the future
OK
Cancel
Help
7:44:45 PM Executing tasks: [:FtcRobotContr
7:44:55 PM Gradle build finished in 17s 88

What Not To Do



Wireless ADB Setup and Demo

Justin Jiang

Overview

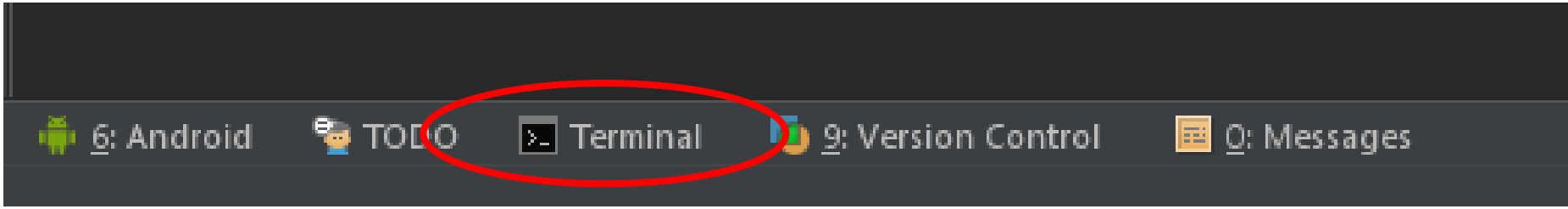
- What is ADB?
- Setup Wireless ADB between Android Studio and Robot Controller Phone
- Workflow with wireless ADB
- Live Demo

Android Debug Bridge (ADB)

- The bridge between the Android Studio and Android Devices for debugging purpose
- Allow issuing commands within Android Studio to the connected Android Devices and collecting information from Android devices to Android Studio
- USB ADB
- Wireless ADB
  - Using home wireless router
  - Using laptop as hotspot

Setup Wireless ADB

1. Go to the ‘Terminal’ tab at the bottom of Android Studio



Setup Wireless ADB in Android Studio

2. Issue command: “adb kill-server”
3. Issue command: “adb usb” (Should see message ‘restarting in USB mode’ and phone appearing in the list)
4. Issue command: “adb tcpip 5555”
5. Go to your phone to find out your phone’s ip address on the home network, Settings > Wifi > “Connected Wifi” > IP Address
6. Issue command: “adb connect <IP Address>”
7. Unplug your usb cable
8. Issue command: “adb devices” to confirm your Android device via wireless connection is listed
9. Your device should also been seen in ‘Android’ tab at the bottom of Android Studio

Workflow with Wireless ADB

1. At beginning of each debug session, set up the wireless ADB between Android Studio and the Robot Controller Phone based on the instructions in the previous slide
2. Pair Robot Controller Phone with Driver Station phone in the normal procedure
3. Write/Update your program and run the app from Android Studio, select the Robot Controller Phone on the wireless ADB created earlier
4. The pairing will be interrupted, the FtcRobotControllerApp will be updated and restarted, upon restarting, the pairing with Driver station phone is automatically resumed
5. Test your program with your Driver Station phone (repeat Steps 3-5 as many times as you need in the debugging session)

Live Demo