How to use GitHub

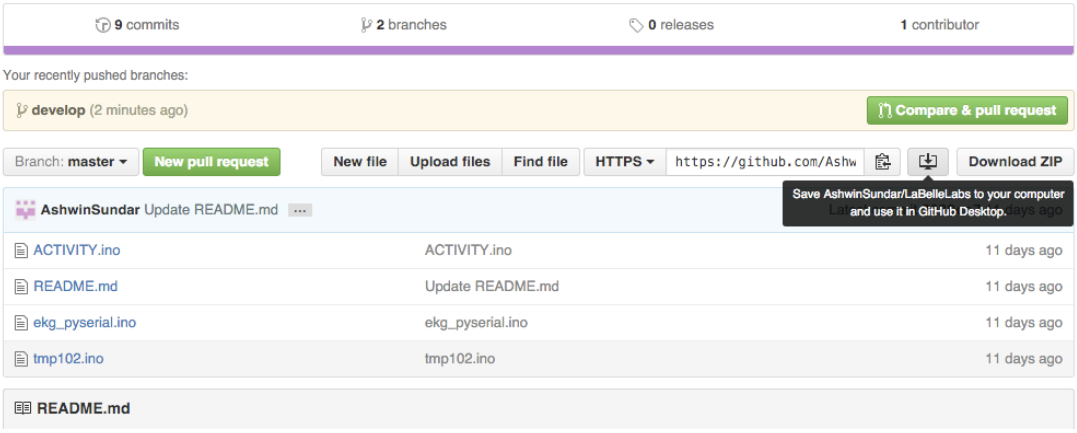
by Ashwin Sundar

GitHub is a repository hosting service that offers distributed revision control and source code management (SCM). My goal is to reduce the amount of "cowboy engineering" we are forced to do. I want us to make incremental changes and steady progress toward the goal - a piece of software that stores and transmits temperature, EKG, and activity data. GitHub helps us make this development process modular and parallelize the workflow - each of us can take responsibility for a section of the code, and work on it in a cooperative manner.

Please note: our repository is privately editable, but publicly viewable. I don't anticipate this to be a problem, but I'm not sure if LaBelle Labs includes software code in their intellectual property. Alejo, if you feel this is an issue, let me know and we can explore other options.

Setting up GitHub

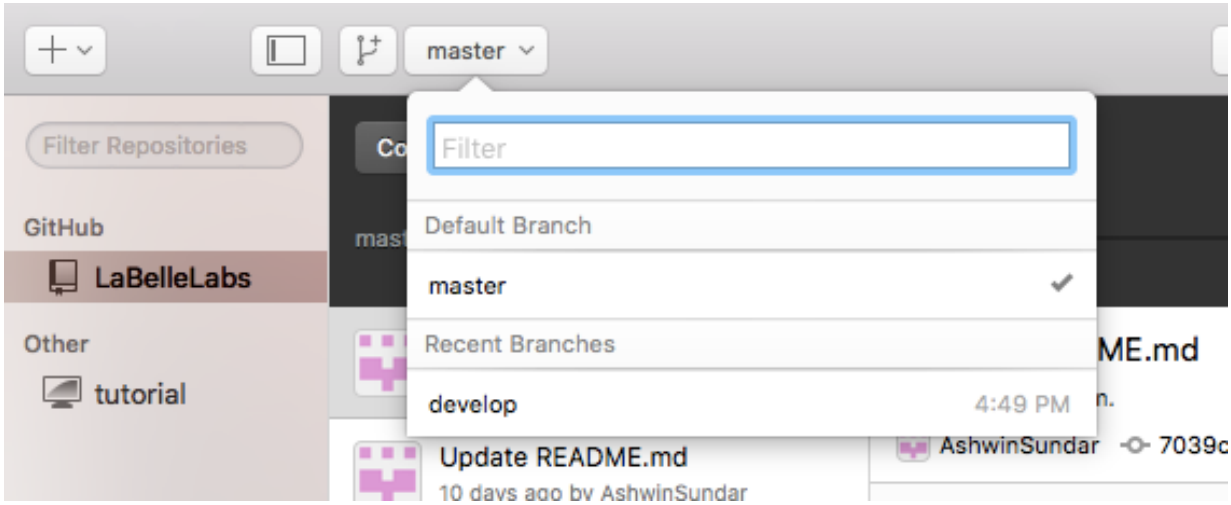
1. **Download the desktop client at https://desktop.github.com/**
2. **Create an account on github.com**
3. **Navigate to https://github.com/AshwinSundar/LaBelleLabs**
4. **Click on "Save AshwinSundar/LaBelleLabs to your computer and use it in GitHub Desktop."**



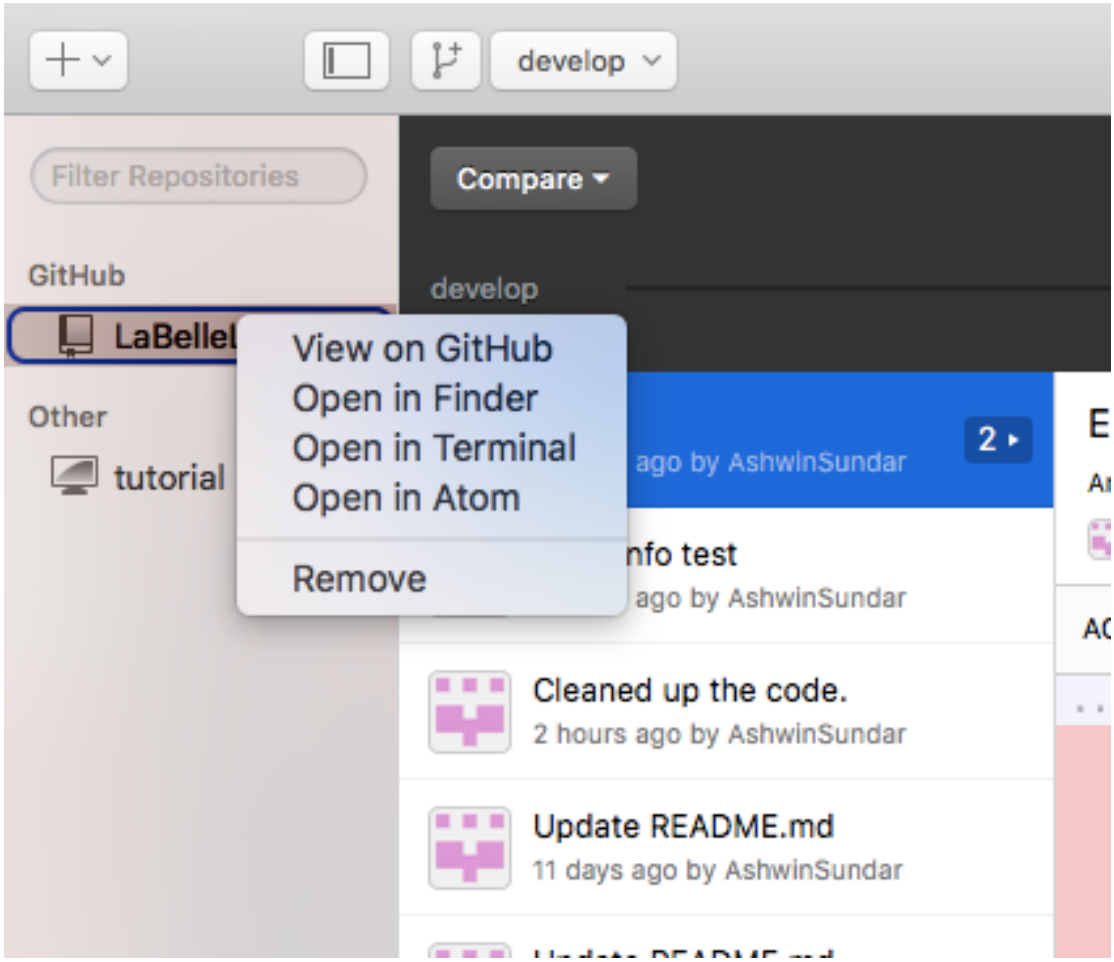
5. **If a dialogue pops up, click "Launch Application"**
6. **You're done! You are now participating in version control, an important part of software development. When you have multiple developers changing the same piece of code, it's important to track changes and allow developers to revert back to functional code at any time.**

Editing your first file

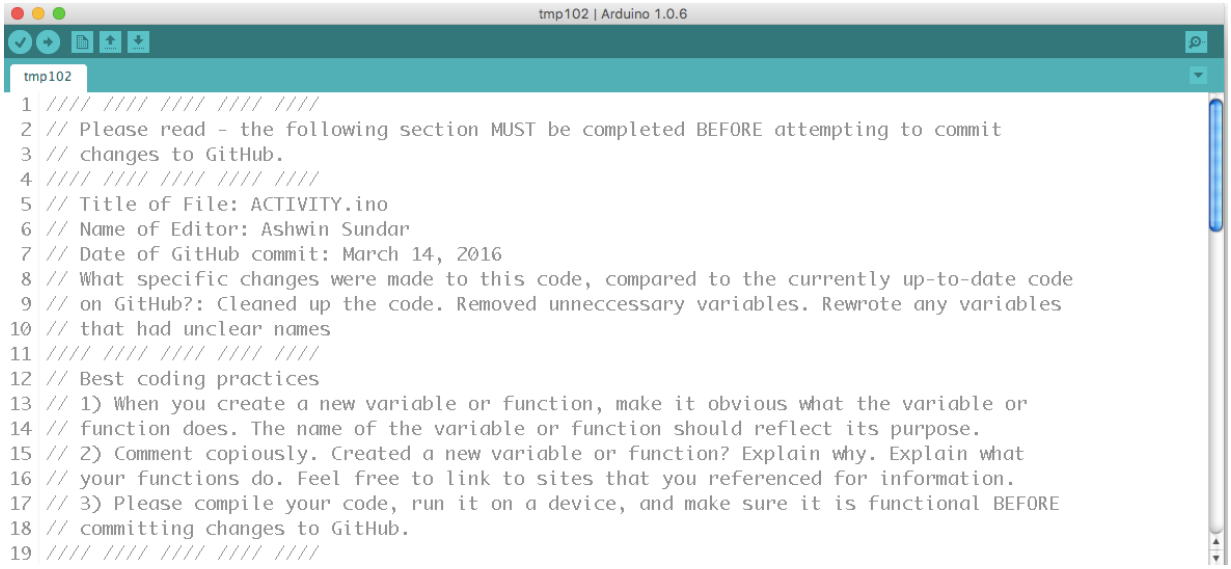
1. **In GitHub Desktop, change your current branch to “develop”. You are only permitted to make changes in this branch.**



2. **Right click on “LaBelleLabs” in the Repositories Menu. Select “Open in Finder” if you’re on a Mac. On a PC, it may read something like “Open folder”.**



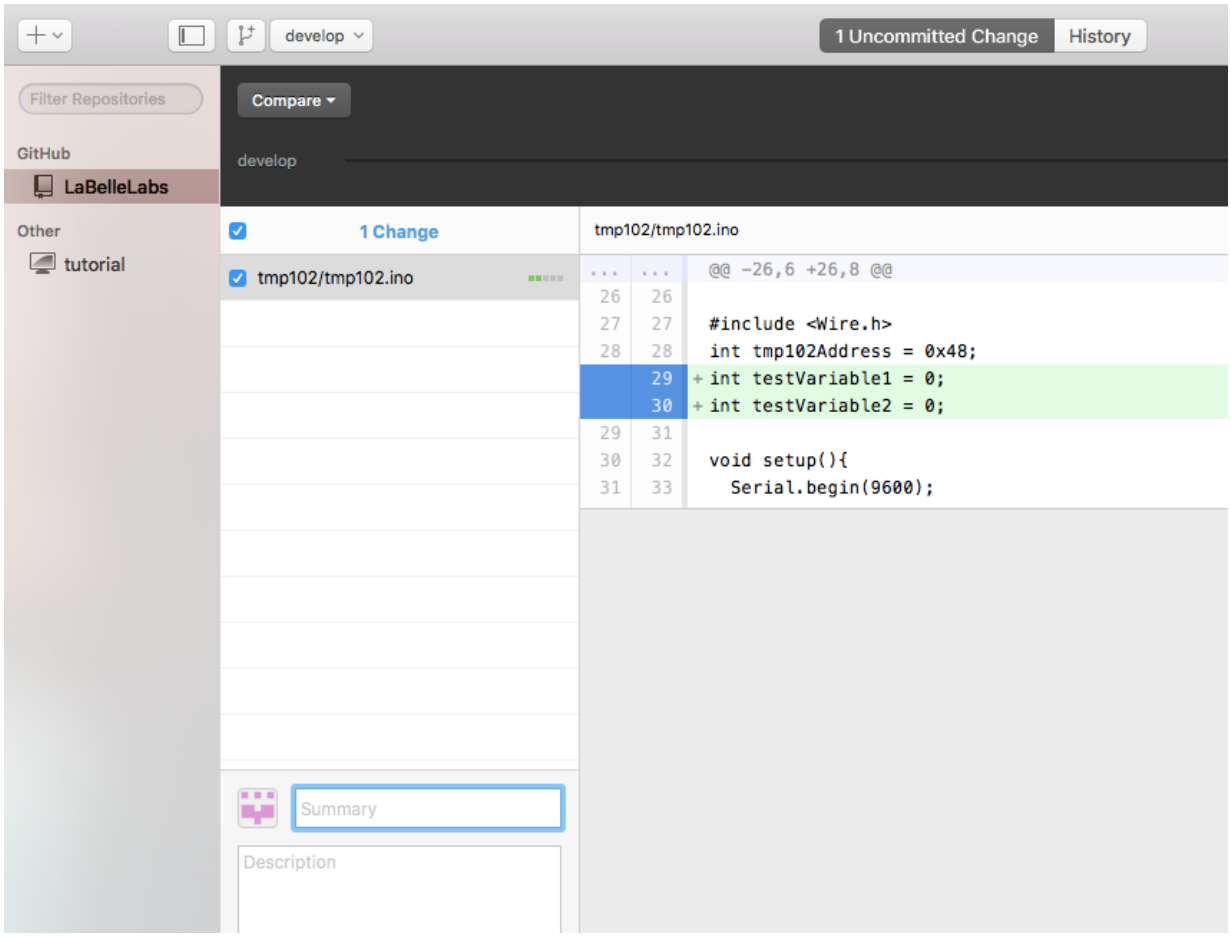
3. **From here, you are taken to a Local Git Repository. GitHub automatically links files in this folder with the LaBelleLabs online repository. You are NOT permitted to add new files to this folder.**
4. **Let's edit tmp102.ino. Double click on tmp102.ino to open it in the Arduino IDE.**



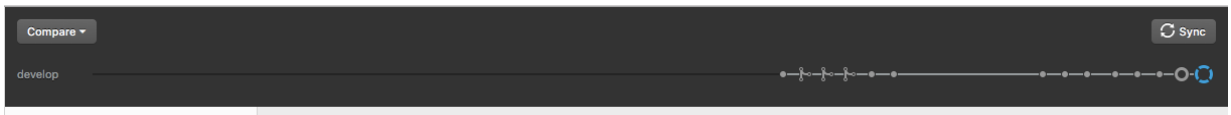
5. **Take a moment to read the header section I've written at the top of tmp102.ino (as well as ACTIVITY.ino and ekg_pyserial.ino). Commenting your code is very important in a collaborative environment.**
6. **Let's make an edit. Create some test variables starting at line 29.**

```
21 ////////////////////////////////////////////////////////////////////
22 //©2011 bildr
23 //Released under the MIT License - Please reuse change and share
24 //Simple code for the TMP102, simply prints temperature via serial
25 ////////////////////////////////////////////////////////////////////
26
27 #include <Wire.h>
28 int tmp102Address = 0x48;
29 int testVariable1 = 0;
30 int testVariable2 = 0;
31
32 void setup(){
33   Serial.begin(9600);
34   Wire.begin();
35   Serial.println("reset");
36 }
```

7. **Save the file.**
8. **Go back to the GitHub Desktop client. Notice how you now have an uncommitted change. The great thing about GitHub is it's really easy to see exactly what changes you've made.**



10. **Let's commit our first change. In "Summary", write "How to use GitHub practice edit". In "Description", write "Added two new test variables. I will store baseline values for the threshold function in these variables."**
11. **Click "Commit to develop".**
12. **You've successfully committed your first change! Now let's sync ourselves with the server, to make sure we have the latest files. The blue circle represents your current location. Filled-in dots represent changes that have been synced with your computer. Open circles represents changes that have NOT been synced. Notice that we are out of sync. Click "Sync" in the top right.**



13. **Now you are up to date! You are all set to develop software for LaBelleLabs now. Remember - we are all responsible for source code management. Everyone must take an active role in commenting their code meaningfully and clearly. It's a good practice to press the "Sync" button in the top right of the GitHub Desktop client periodically (and ESPECIALLY when you open up GitHub Desktop for the first time in the morning!). You never know when**

someone has submitted a change!