Electrical Training

Week 3: GitHub and “Breadboard Arduino” Prototyping

Agenda

- GitHub
- “Breadboard Arduino” Prototyping

GitHub

GitHub

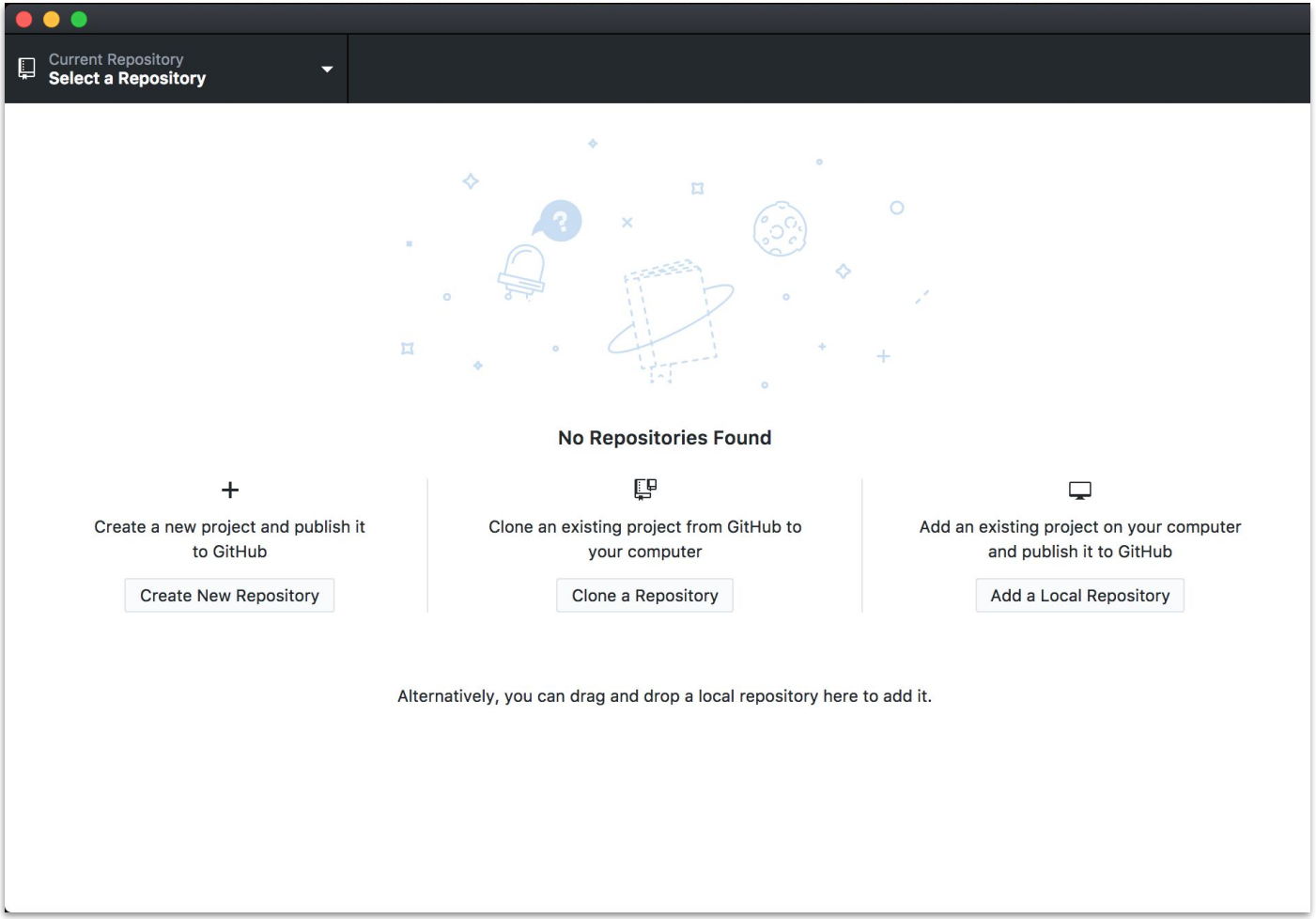
GitHub

- Version control service that uses Git
  - Tracks changes made to files in a **repository**
- You can update the version of the files locally by **committing** them
- You can prototype by creating a **branch** from a project
- Once you are done testing your branch, you can **merge** it with the **master branch** (this is known as a **pull** request)

Getting Started

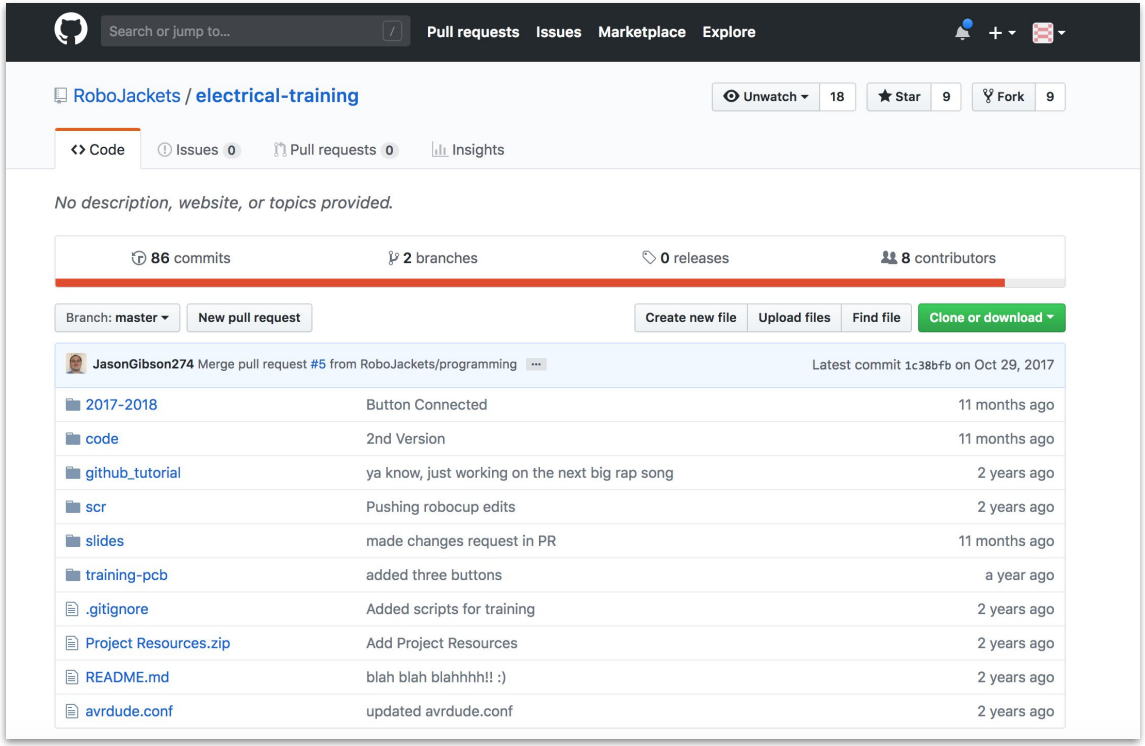
- Sign Up
  - https://github.com/join
- Download Desktop Client
  - https://desktop.github.com

Desktop Client



Navigate to the Repo

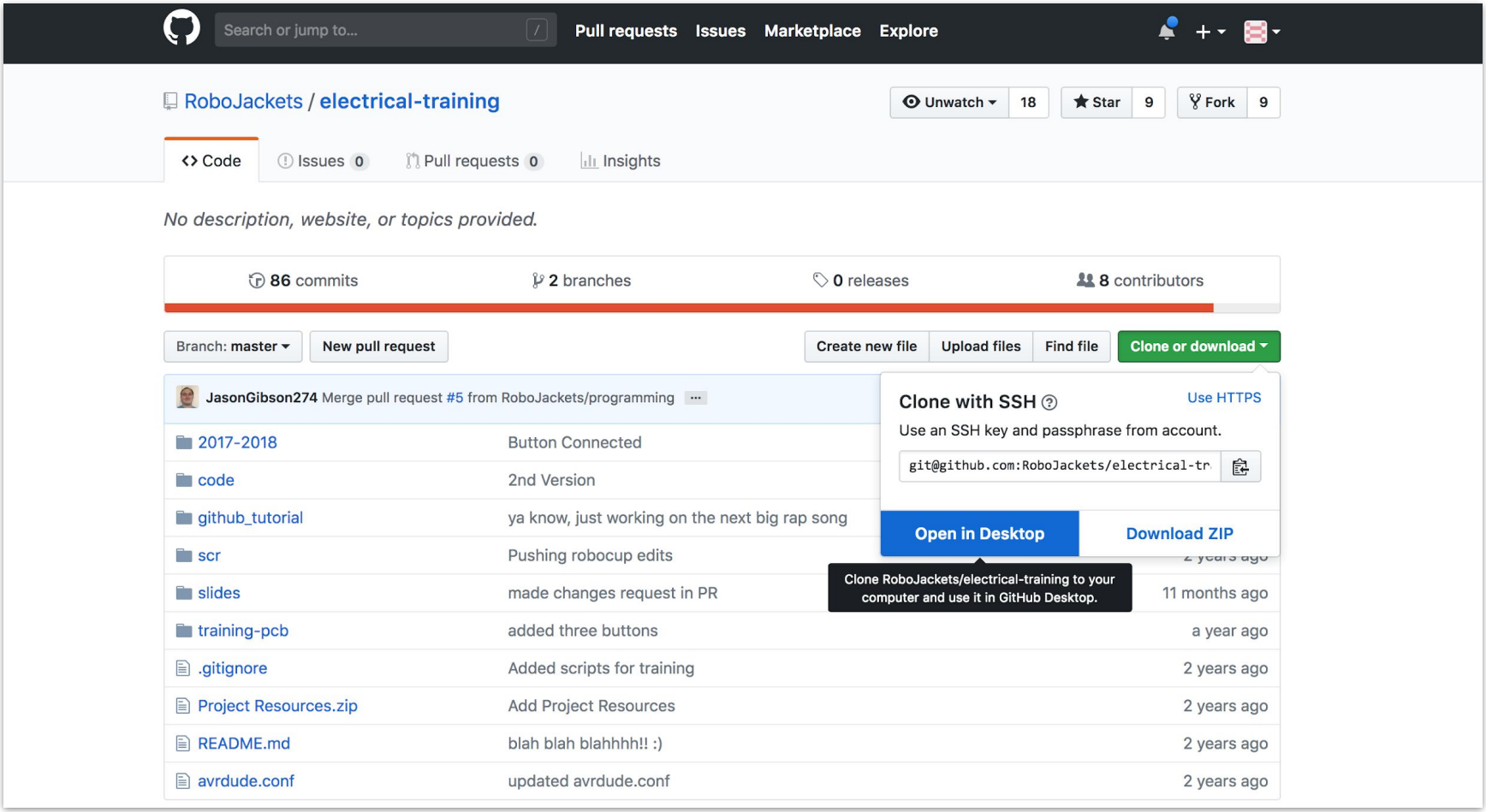
https://github.com/RoboJackets/electrical-training



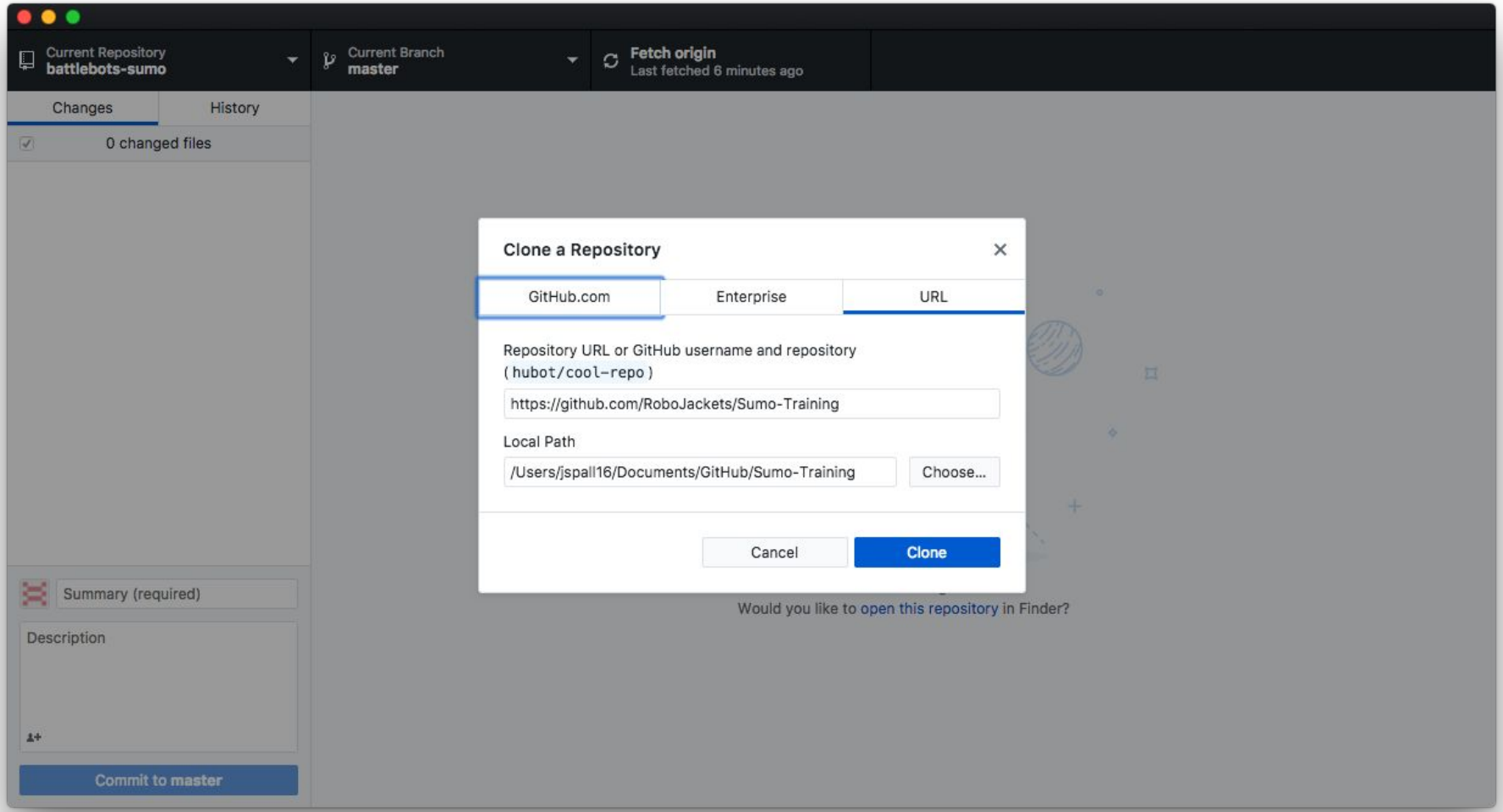
Cloning

- Copies the repository to your local machine
- Go to the repo and click the “Clone or Download” button
  - Should bring up the Desktop Client
- You have to choose a file path for your files to be cloned to

Cloning to Desktop App



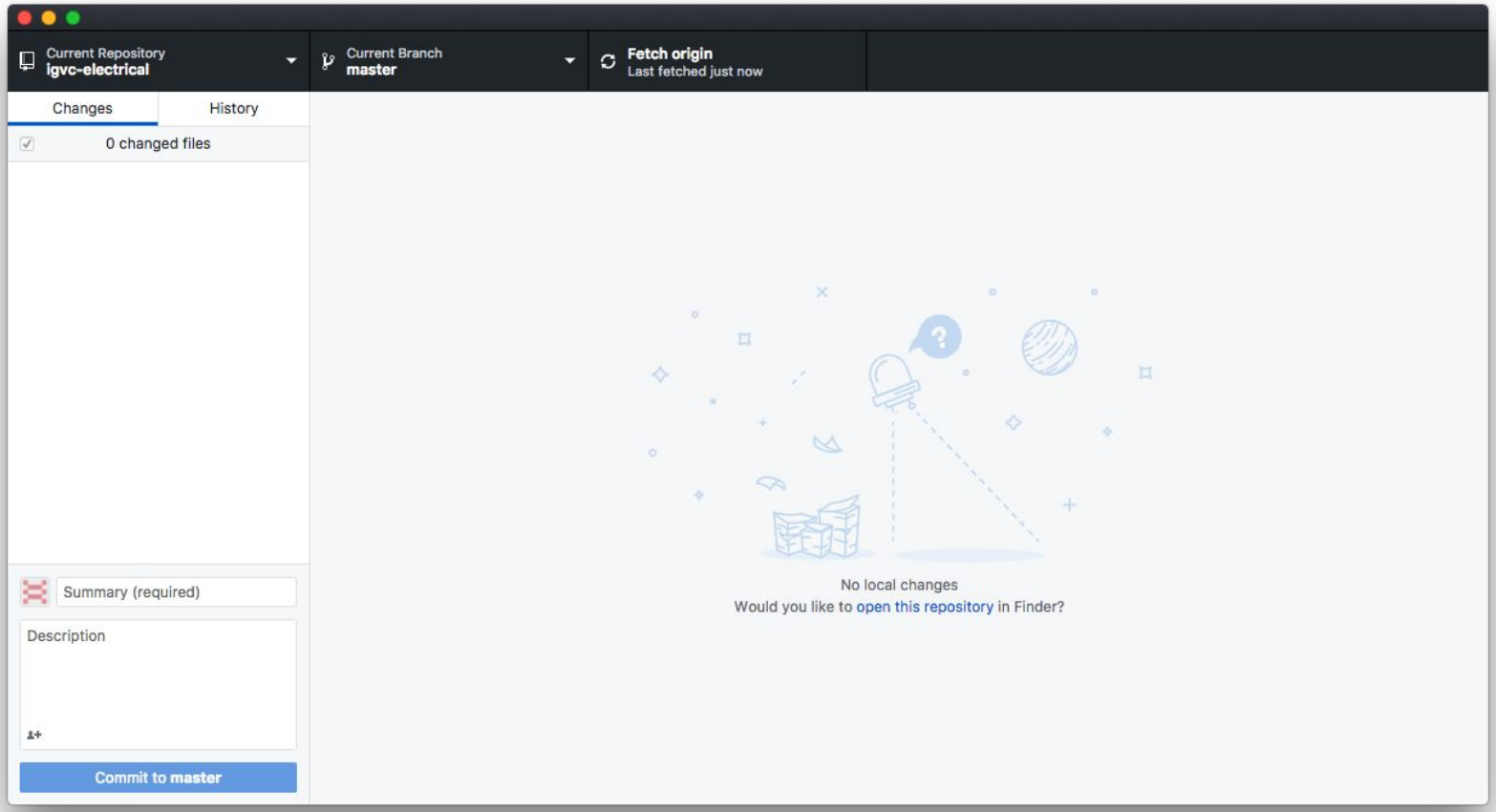
Setting File Path



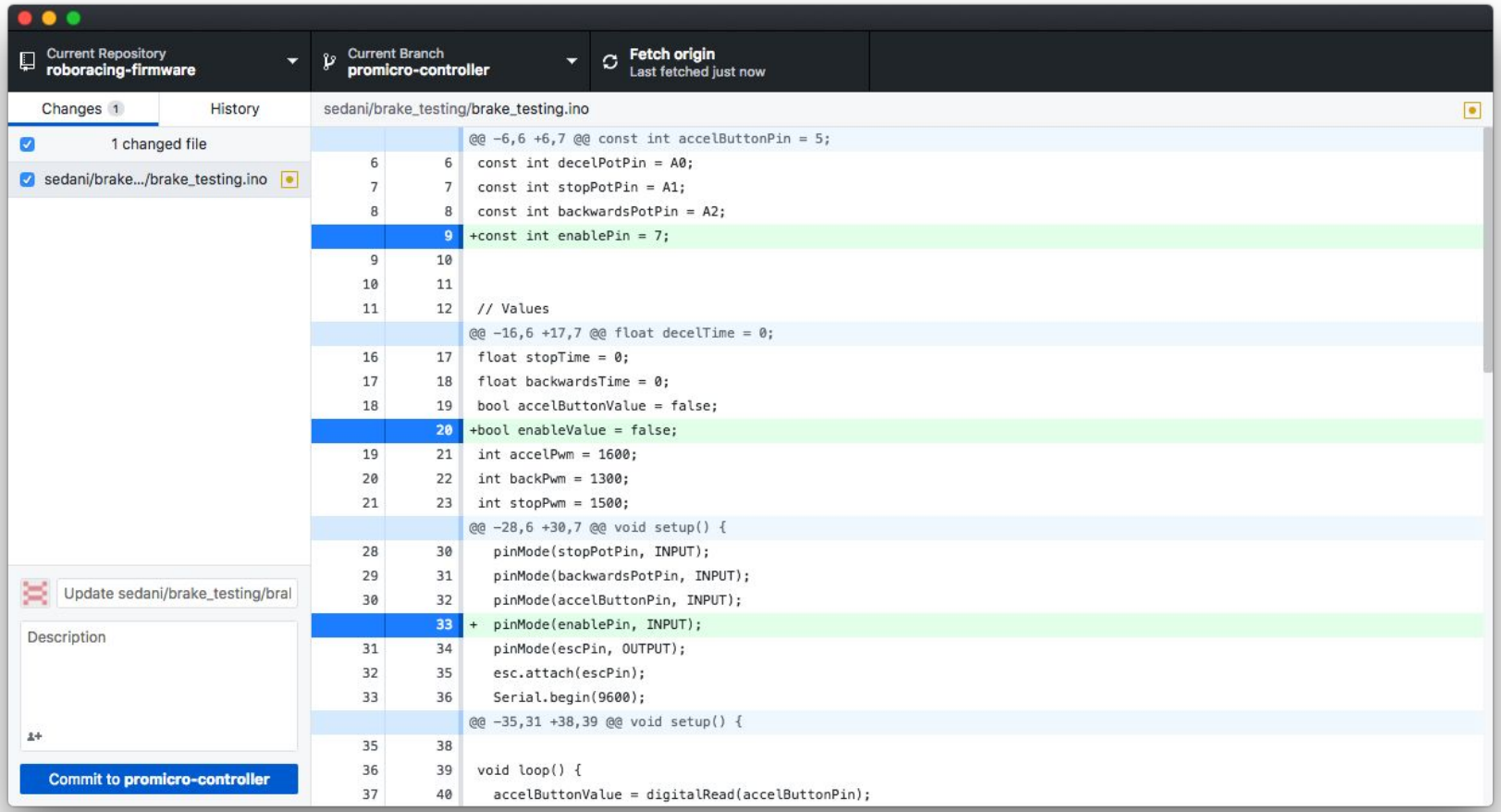
Navigating the Repo

- Once cloned, the desktop client will show either of two views
- “Changes”
  - The currently modified elements of your files on your local machine
  - Lines that are green are additions
  - Lines that are red are removals
- “History”
  - The previous commits and content of changes

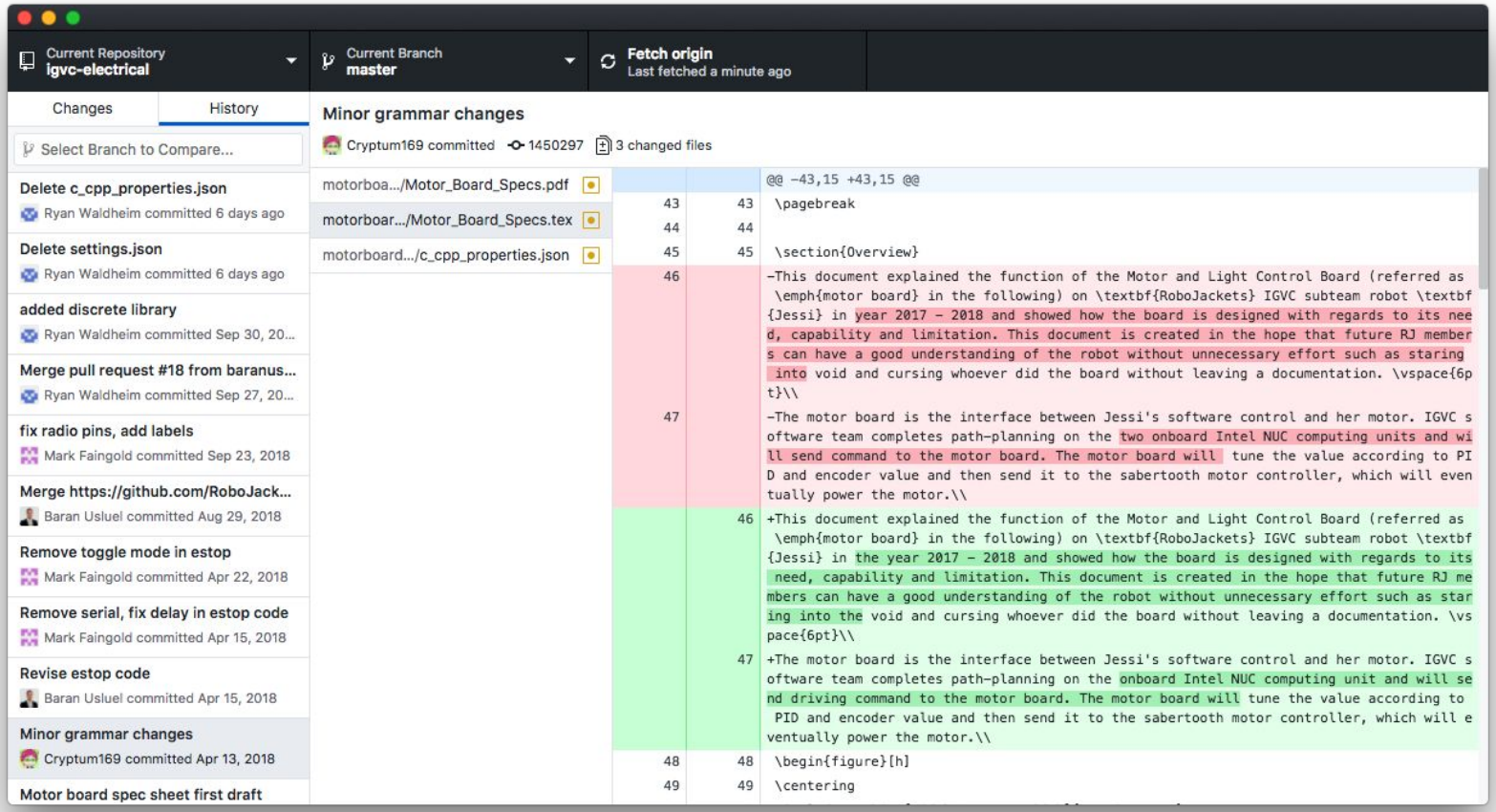
No Current Changes



Changes


Current Repository
roboracing-firmware
Current Branch
promicro-controller
Fetch origin
Last fetched just now
Changes 1
History
1 changed file
sedani/brake.../brake_testing.ino
Update sedani/brake_testing/bral
Description
Commit to promicro-controller
sedani/brake_testing/brake_testing.ino
@@ -6,6 +6,7 @@ const int accelButtonPin = 5;
6 6 const int decelPotPin = A0;
7 7 const int stopPotPin = A1;
8 8 const int backwardsPotPin = A2;
9 +const int enablePin = 7;
9 10
10 11
11 12 // Values
@@ -16,6 +17,7 @@ float decelTime = 0;
16 17 float stopTime = 0;
17 18 float backwardsTime = 0;
18 19 bool accelButtonValue = false;
20 +bool enableValue = false;
19 21 int accelPwm = 1600;
20 22 int backPwm = 1300;
21 23 int stopPwm = 1500;
@@ -28,6 +30,7 @@ void setup() {
28 30 pinMode(stopPotPin, INPUT);
29 31 pinMode(backwardsPotPin, INPUT);
30 32 pinMode(accelButtonPin, INPUT);
33 + pinMode(enablePin, INPUT);
31 34 pinMode(escPin, OUTPUT);
32 35 esc.attach(escPin);
33 36 Serial.begin(9600);
@@ -35,31 +38,39 @@ void setup() {
35 38
36 39 void loop() {
37 40 accelButtonValue = digitalRead(accelButtonPin);

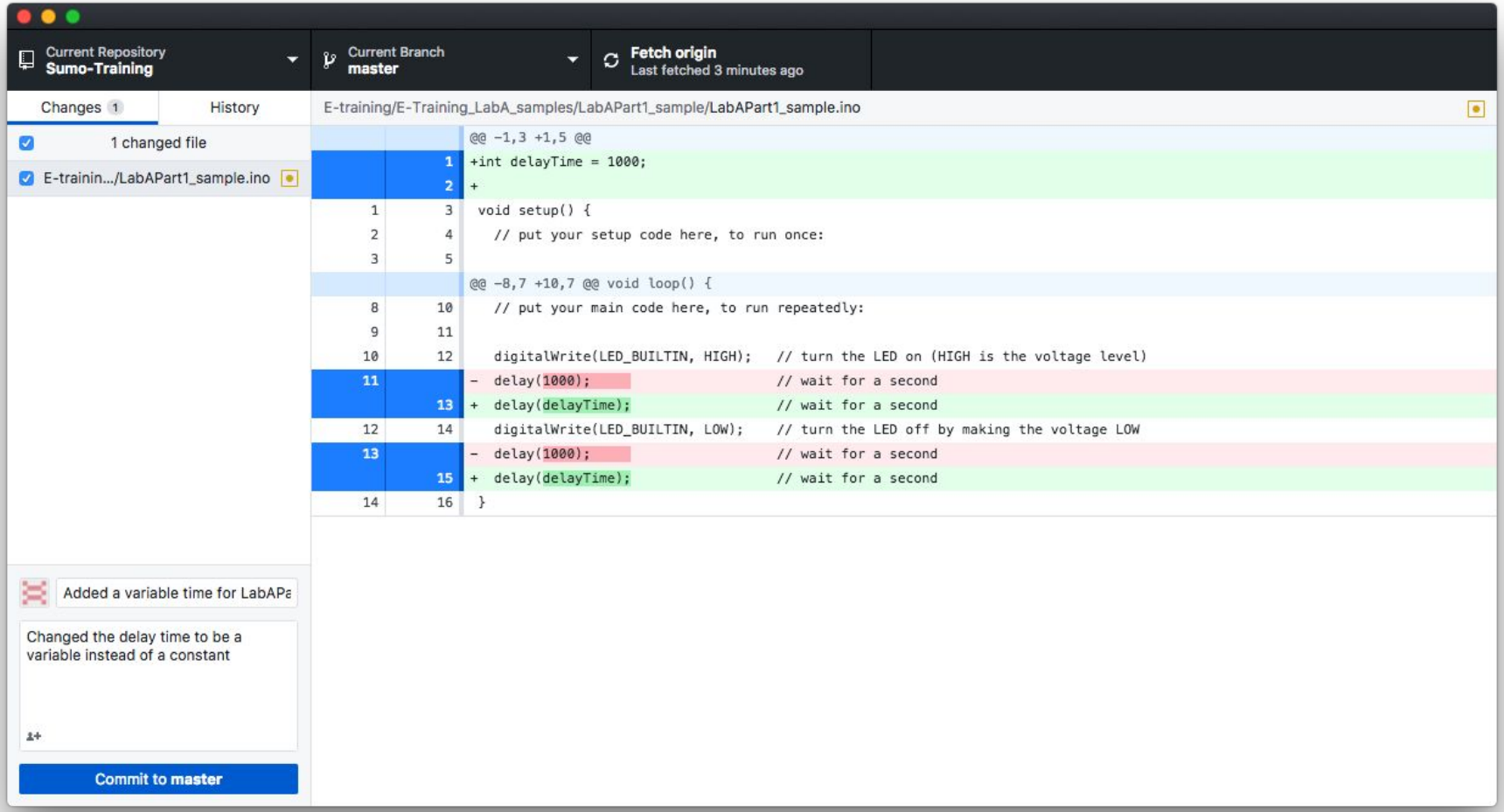
History


Current Repository
igvc-electrical
Current Branch
master
Fetch origin
Last fetched a minute ago
Changes
History
Select Branch to Compare...
Delete c_cpp_properties.json
Ryan Waldheim committed 6 days ago
Delete settings.json
Ryan Waldheim committed 6 days ago
added discrete library
Ryan Waldheim committed Sep 30, 20...
Merge pull request #18 from baranus...
Ryan Waldheim committed Sep 27, 20...
fix radio pins, add labels
Mark Faingold committed Sep 23, 2018
Merge https://github.com/RoboJack...
Baran Usluel committed Aug 29, 2018
Remove toggle mode in estop
Mark Faingold committed Apr 22, 2018
Remove serial, fix delay in estop code
Mark Faingold committed Apr 15, 2018
Revise estop code
Baran Usluel committed Apr 15, 2018
Minor grammar changes
Cryptum169 committed Apr 13, 2018
Motor board spec sheet first draft
Minor grammar changes
Cryptum169 committed
1450297
3 changed files
motorboa.../Motor_Board_Specs.pdf
motorboar.../Motor_Board_Specs.tex
motorboard.../c_cpp_properties.json
@@ -43,15 +43,15 @@
\pagebreak
\section{Overview}
-This document explained the function of the Motor and Light Control Board (referred as \emph{motor board} in the following) on \textbf{RoboJackets} IGVC subteam robot \textbf{Jessi} in year 2017 - 2018 and showed how the board is designed with regards to its need, capability and limitation. This document is created in the hope that future RJ members can have a good understanding of the robot without unnecessary effort such as staring into void and cursing whoever did the board without leaving a documentation. \vspace{6pt}\\
-The motor board is the interface between Jessi's software control and her motor. IGVC software team completes path-planning on the two onboard Intel NUC computing units and will send command to the motor board. The motor board will tune the value according to PID and encoder value and then send it to the sabertooth motor controller, which will eventually power the motor.\\
+This document explained the function of the Motor and Light Control Board (referred as \emph{motor board} in the following) on \textbf{RoboJackets} IGVC subteam robot \textbf{Jessi} in the year 2017 - 2018 and showed how the board is designed with regards to its need, capability and limitation. This document is created in the hope that future RJ members can have a good understanding of the robot without unnecessary effort such as staring into the void and cursing whoever did the board without leaving a documentation. \vspace{6pt}\\
+The motor board is the interface between Jessi's software control and her motor. IGVC software team completes path-planning on the onboard Intel NUC computing unit and will send driving command to the motor board. The motor board will tune the value according to PID and encoder value and then send it to the sabertooth motor controller, which will eventually power the motor.\\
\begin{figure}[h]
\centering

Editing

- Before you start editing, remember to Pull the newest version of the repository
  - Button in the top right with the sync icon
- Edit the files like normal
-

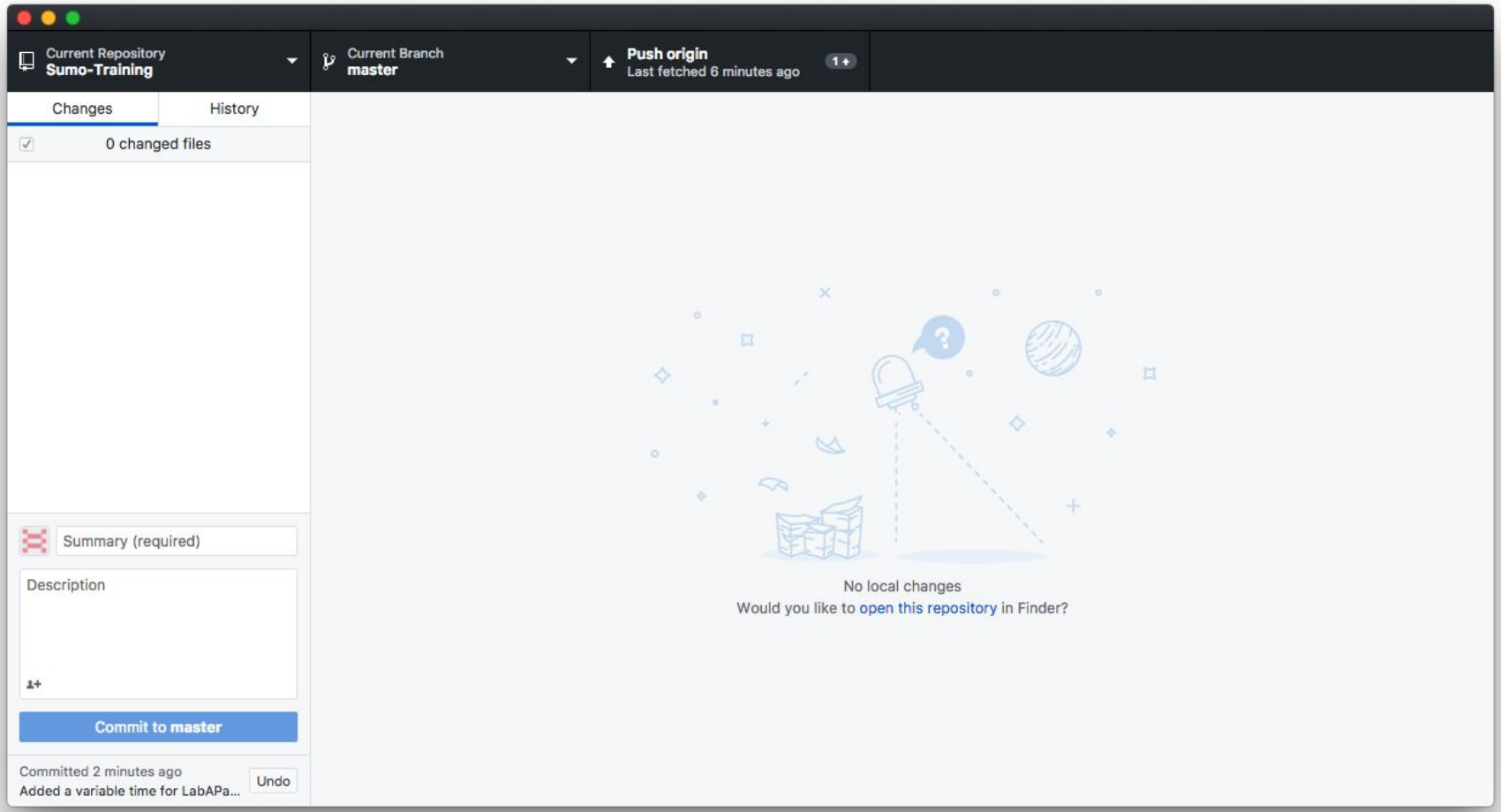
Edits


Current Repository
Sumo-Training
Current Branch
master
Fetch origin
Last fetched 3 minutes ago
Changes 1
History
1 changed file
E-trainin.../LabAPart1_sample.ino
E-training/E-Training_LabA_samples/LabAPart1_sample/LabAPart1_sample.ino
@@ -1,3 +1,5 @@
+int delayTime = 1000;
+
void setup() {
// put your setup code here, to run once:
@@ -8,7 +10,7 @@ void loop() {
// put your main code here, to run repeatedly:
digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)
- delay(1000); // wait for a second
+ delay(delayTime); // wait for a second
digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW
- delay(1000); // wait for a second
+ delay(delayTime); // wait for a second
}
Added a variable time for LabAPa
Changed the delay time to be a variable instead of a constant
Commit to master

Committing and Pushing

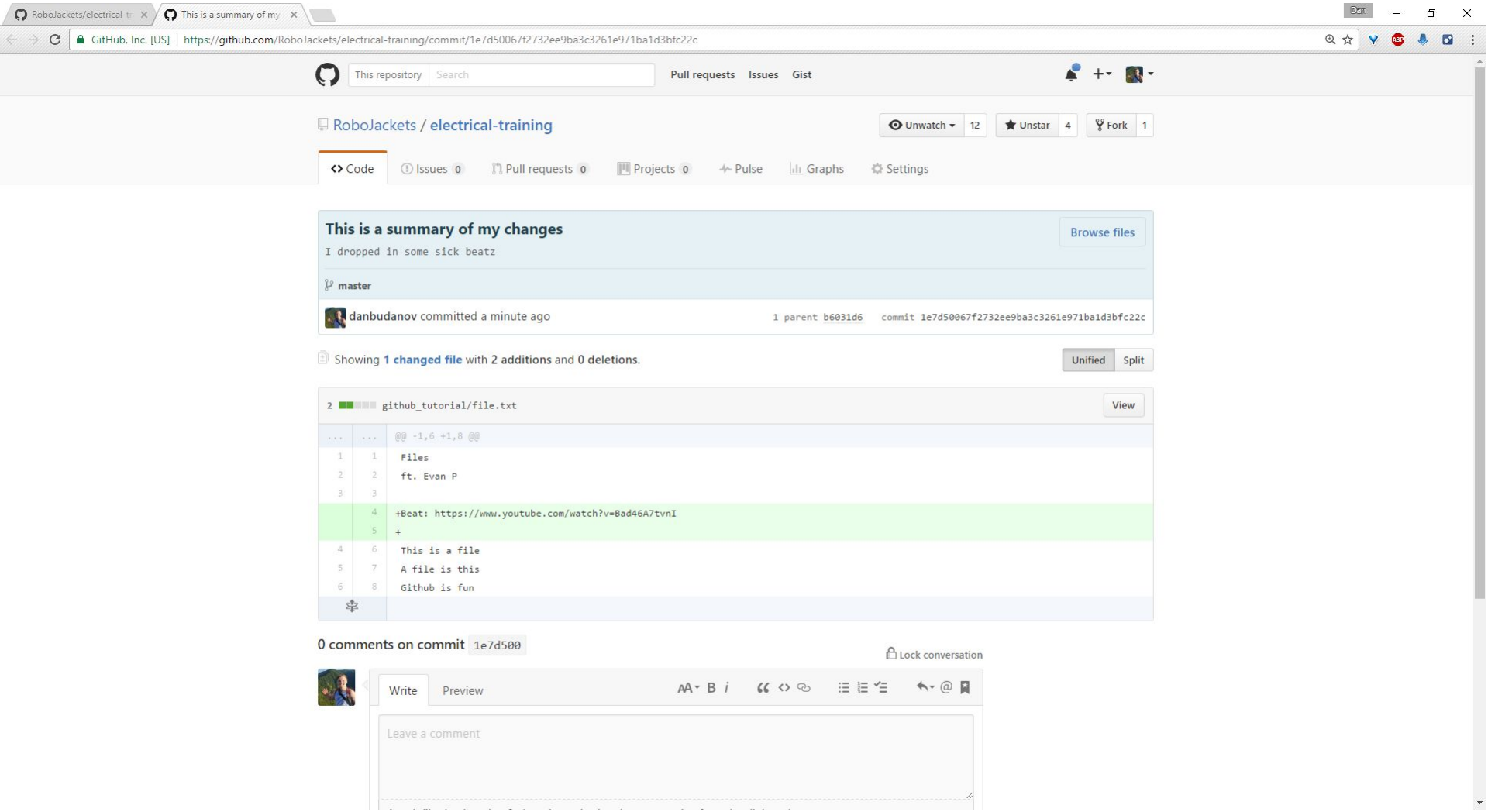
- Once finished editing, commit the changes
  - Click the blue “Commit” button on the bottom of the screen
  - Remember to give a summarizing title and a description of the changes you have made
- After commiting, push the changes to the online repository
  - Button in the top right that says “Push…”

Pushing


Current Repository
Sumo-Training
Current Branch
master
Push origin
Last fetched 6 minutes ago
1
Changes
History
0 changed files
Summary (required)
Description
Commit to master
Committed 2 minutes ago
Added a variable time for LabAPa...
Undo
No local changes
Would you like to open this repository in Finder?

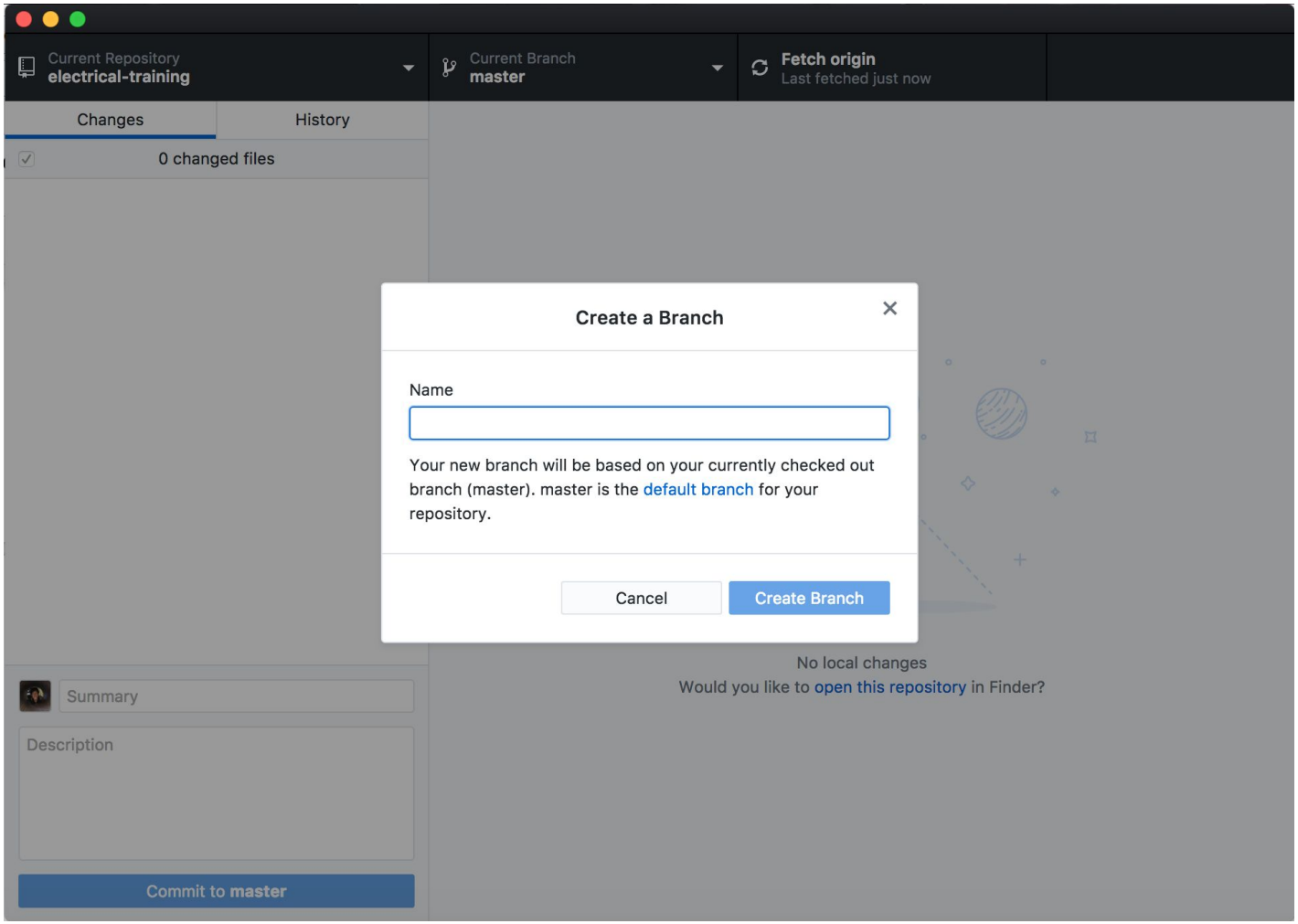
Result

- After pushing the changes, they will appear in the branch you were editing
- Everyone else can pull your current changes


RoboJackets/electrical-tr
This is a summary of my
Dan
GitHub, Inc. [US] | https://github.com/RoboJackets/electrical-training/commit/1e7d50067f2732ee9ba3c3261e971ba1d3bfc22c
This repository
Search
Pull requests
Issues
Gist
RoboJackets / electrical-training
Unwatch 12
Unstar 4
Fork 1
Code
Issues 0
Pull requests 0
Projects 0
Pulse
Graphs
Settings
This is a summary of my changes
I dropped in some sick beatz
Browse files
master
danbudanov committed a minute ago
1 parent b6031d6
commit 1e7d50067f2732ee9ba3c3261e971ba1d3bfc22c
Showing 1 changed file with 2 additions and 0 deletions.
Unified
Split
2
github_tutorial/file.txt
View
@@ -1,6 +1,8 @@
Files
ft. Evan P
+Beat: https://www.youtube.com/watch?v=Bad46A7tvnI
+
This is a file
A file is this
Github is fun
0 comments on commit 1e7d500
Lock conversation
Write
Preview
Leave a comment

Creating Branches

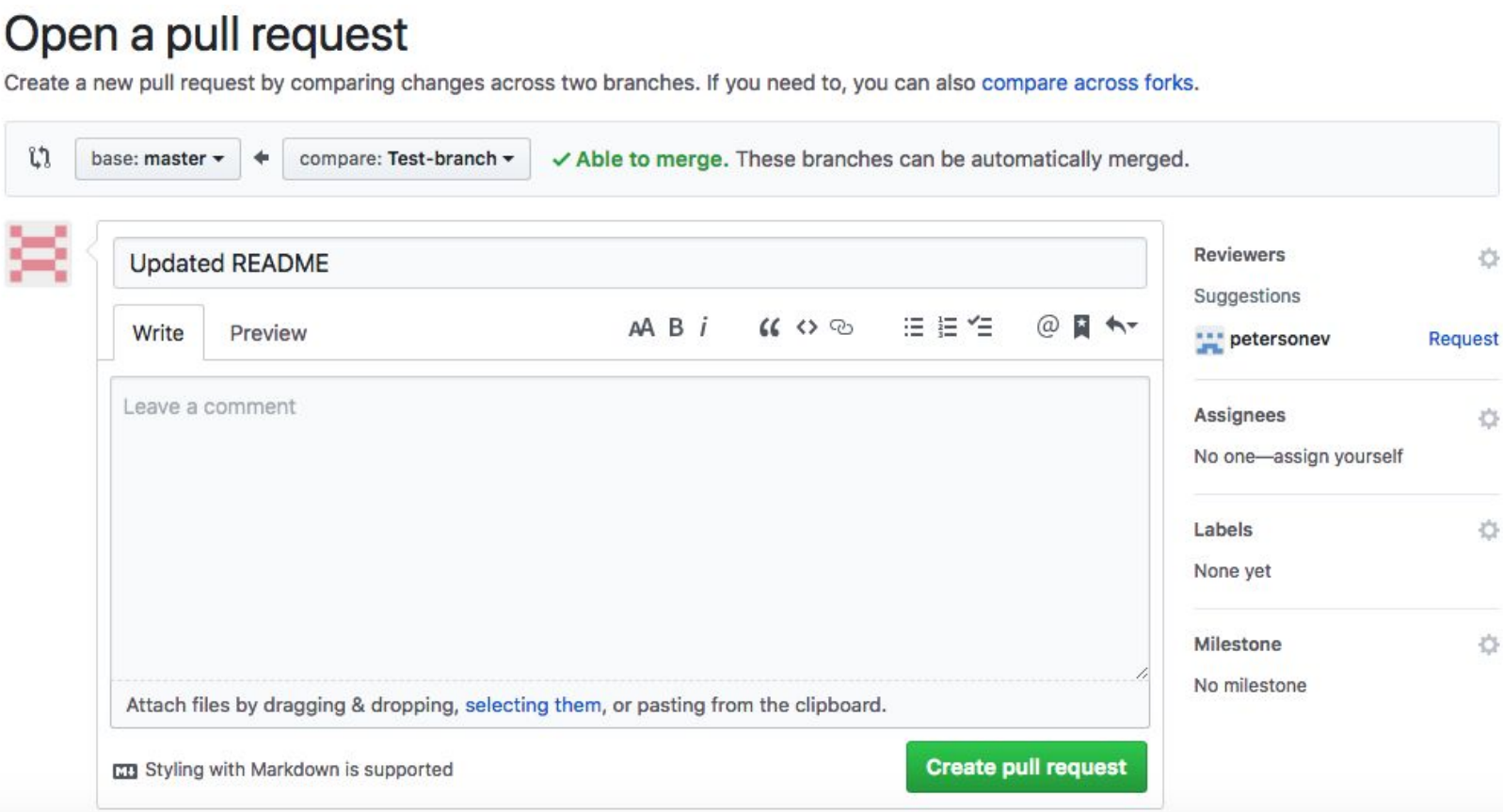
- If someone else is making changes at the same time, you do not want to commit over each other
- Branches let you work off of the same original base and combine changes
  - In the Desktop client, click the middle dropdown and click “New Branch”
  - Choose a name for the branch and click “Create Branch”

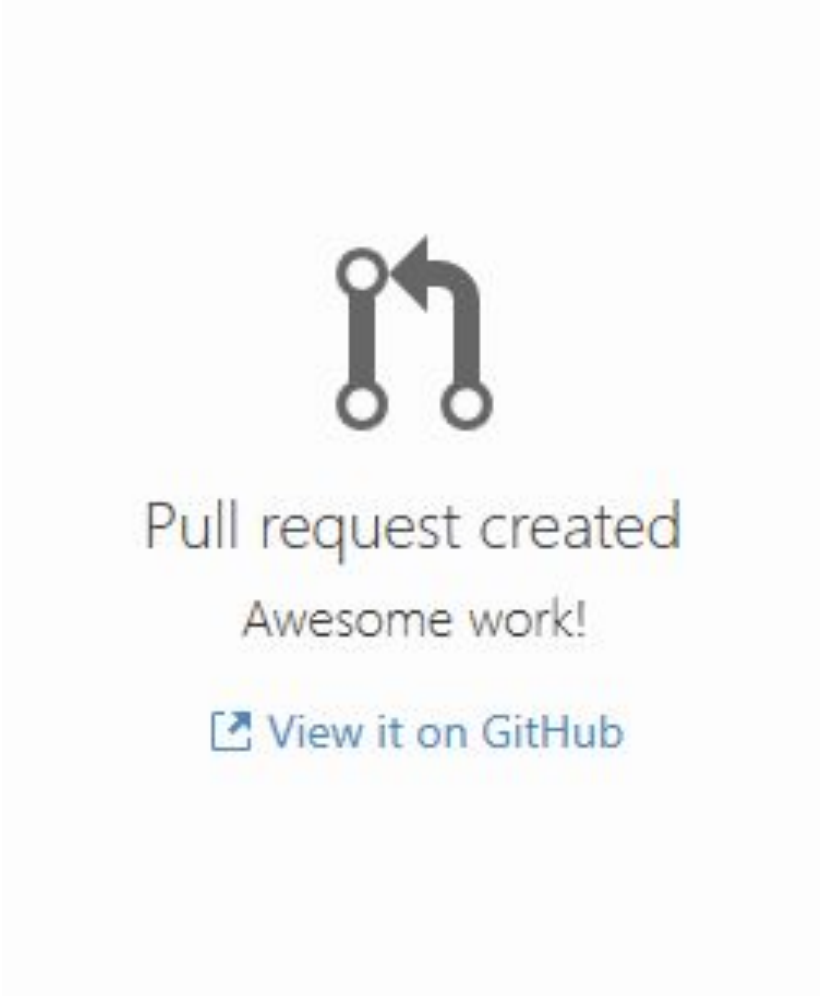

Current Repository
electrical-training
Current Branch
master
Fetch origin
Last fetched just now
Changes
History
0 changed files
Create a Branch
Name
Your new branch will be based on your currently checked out branch (master). master is the default branch for your repository.
Cancel
Create Branch
No local changes
Would you like to open this repository in Finder?
Summary
Description
Commit to master

Pull Requests

- After finishing the changes on your current branch, combine them with “Pull Requests”
- In the web client, click the “New Pull Request” button
- The compare branch is what you are changing
- The base branch is what you are merging into

Example of Pull Request

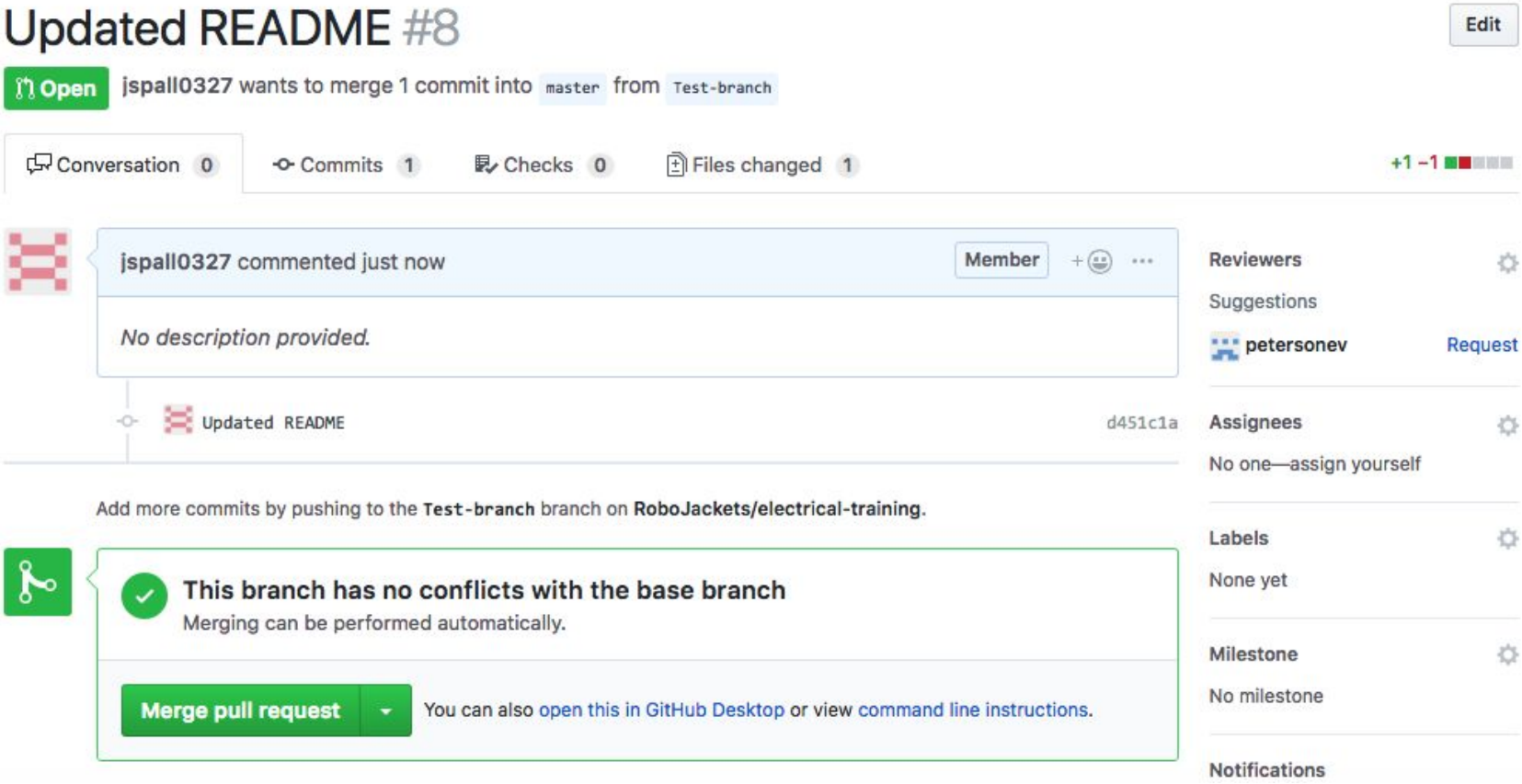



Pull request created
Awesome work!
View it on GitHub

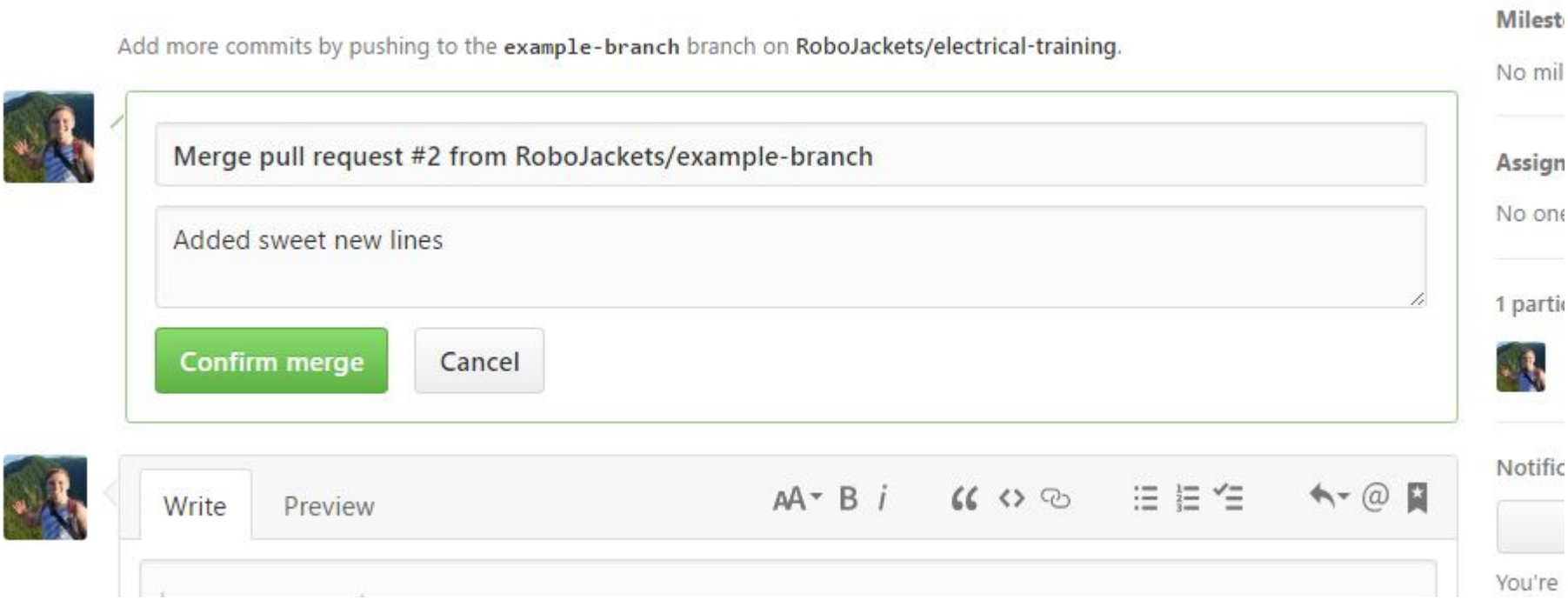
Merging Pull Requests

- After a Pull Request is made, it can be merged into the branch
- Pull Requests are for reviewing, Merging is for the actual combination
- If there are not any conflicts (different changes on the same file), it can be merged and the old branch can be deleted

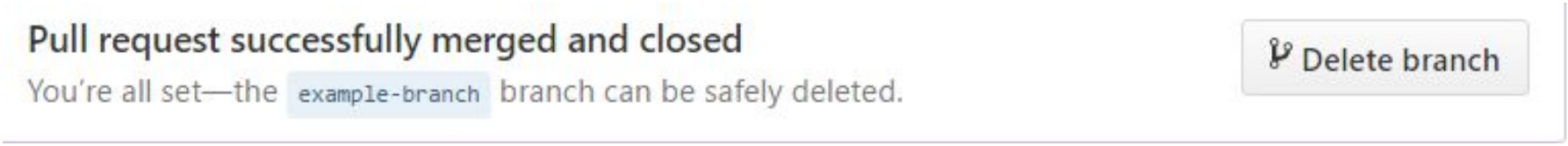
Merging Requests



Final Merge Documentation



Completed Merge and Deleted Branch



“Breadboard Arduino” Prototyping

Worksheet

- Breadboard out a working Arduino using the chips on the official boards
- Follow the worksheet and ask the instructors for assistance