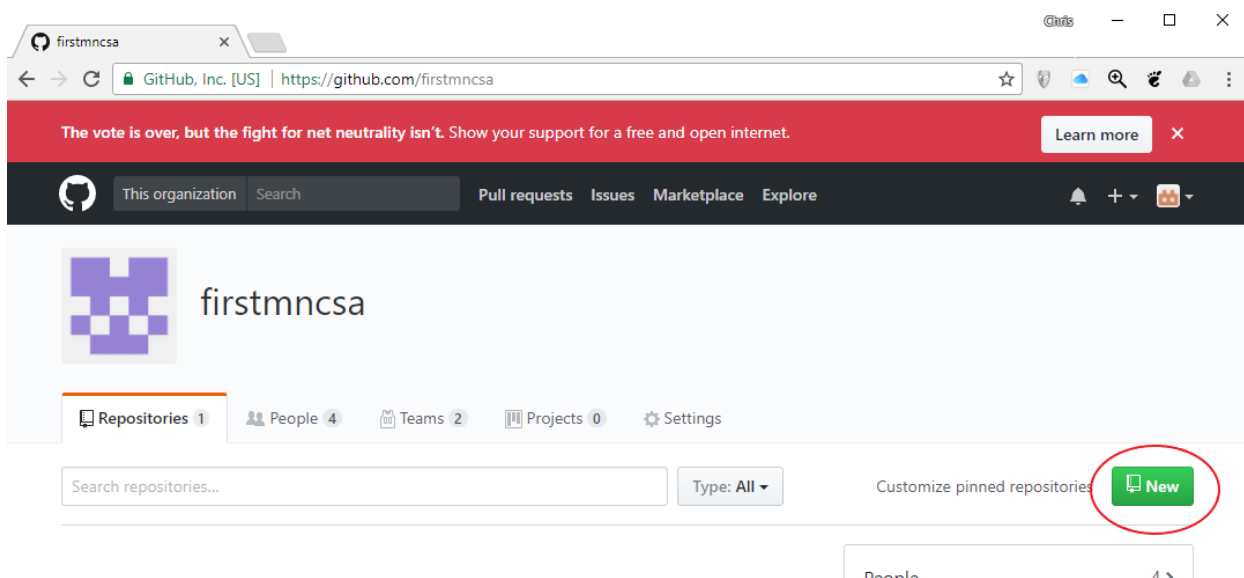


## GitHub Repo Creation.

Before we can post code to GitHub, we need to create a repository. This section assumes you've already created your GitHub accounts and Team for your robotics team. Example shown is a new repo in our MN CSA GitHub team. Also this section assume your account has admin level rights to create repositories for your GitHub team.



1. Login to GitHub and navigate to your team's GitHub repository list.
  - a. Eg; <https://github.com/firstmnscsa>
  - b. Replace firstmnscsa, with your teams GitHub name.
2. Push "New" button on right hand side to create a new repo.



3. Enter the name for the new repository.
  - a. If this repository is for a specific year's robot, it's suggested to use the competition year in the name of the repo for easy identification.
  - b. Eg; powerup\_2018 or frc2018
  - c. If the repository is for a generic code base, a year in the name may not be appropriate.
  - d. Eg; team\_manual or vision\_processing
4. Enter a short description on what the repo will contain.
5. Select if you want the repo to be public.
  - a. Please consider keeping your repo public and encouraging teams to fork your code in the spirit of Gracious Professionalism and they might find a bug in your code and fix it. Or you may even get enhancements from other teams in return!
6. You may want to initialize with a README if you want to write up some documentation for others to read regarding the code in the repo.
  - a. How to use the code,
  - b. Design guidelines.
  - c. Key information needed for use.
  - d. Etc...
7. Push "Create Repository" button when ready to create the repo.

### Create a new repository



A repository contains all the files for your project, including the revision history.

Owner	Repository name
 firstmnscsa	/ powerup2018 

Great repository names are short and memorable. Need inspiration? How about [ideal-sniffle](#).

#### Description (optional)

Robot code for 2018 competition.

- ☒  **Public**  
Anyone can see this repository. You choose who can commit.
- ☐  **Private**  
You choose who can see and commit to this repository.

#### ☐ Initialize this repository with a README

This will let you immediately clone the repository to your computer. Skip this step if you're importing an existing repository.

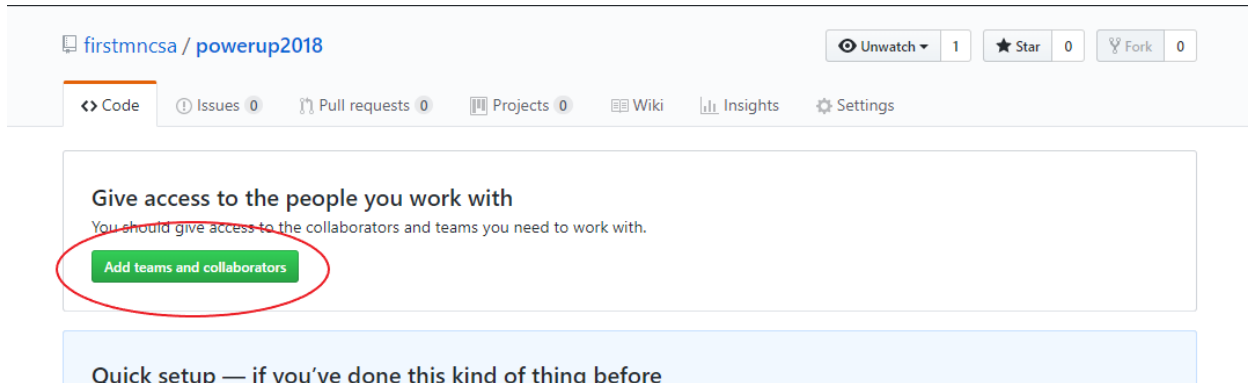
Add .gitignore: **None**

Add a license: **None**

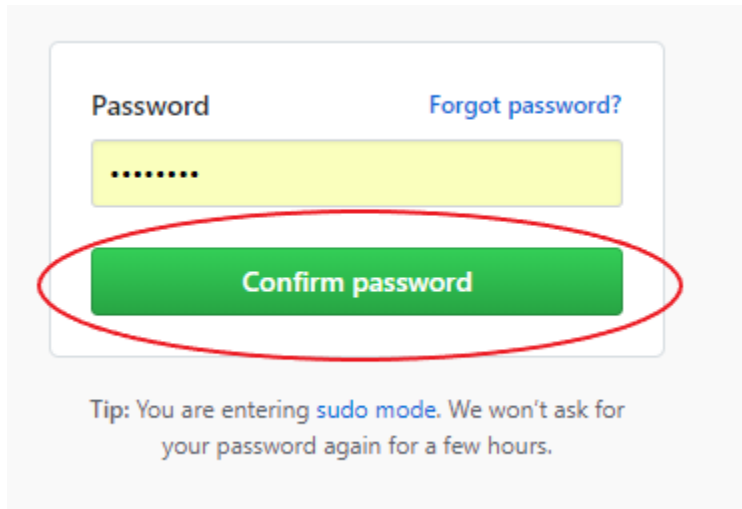


Create repository

8. The next screen allows us to easily add GitHub Teams and Individual account holders to access the repo. Also you can see instructions and URLs for adding the new repo to various different git clients.
9. First up, let's address repo access.
  - a. This is especially critical if you choose to make a private repository as no one will have access to it otherwise.
10. Push "Add teams and collaborators"

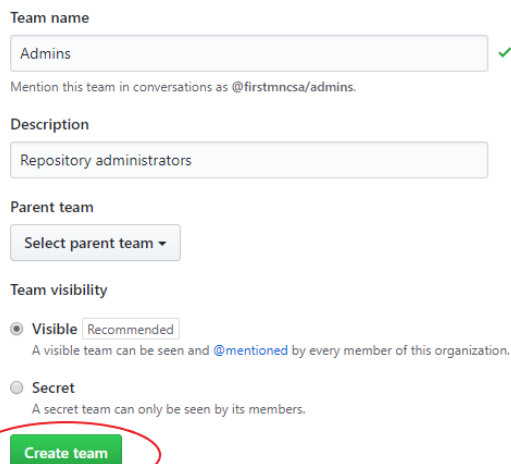


11. You will most likely be challenged to enter your password again, enter it and then push "Confirm Password"

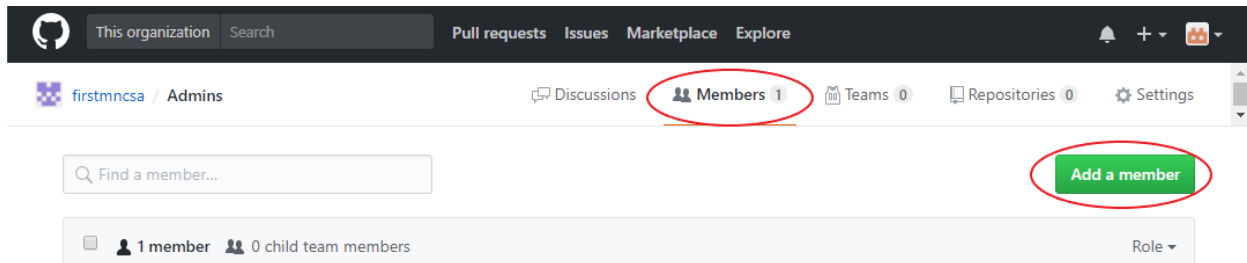


12. Next we need to review, assign and create permissions, teams and individuals who will access and / or modify the repo and it's contents.
13. First up, review the default repository permissions.
  - a. Typically that is going to be set to "write" which allows all individuals you designate on this page to have the ability to change the contents of a repo.
  - b. The default "write" permissions are most likely what you want unless you want to have a single individual or limited individuals with the ability to change repo contents and a lot of folks to have read only access. Note this does not affect public repo which will be read-only to the public by default.
  - c. You can change the default behavior by clicking on "member privileges page"
    - i. Further info on custom access privileges is beyond the scope of this document.
14. Next up is Teams. Teams are groups of individuals that you wish you allocate access rights to as a group vs as an individual.
  - a. Typically you will have a team of "Admins" and another team of "Developers". – this is the model this document will use.
  - b. Individuals can be on either or both teams if that is appropriate.
  - c. Option 1: Create a new team.
    - i. Useful if you don't already have a GitHub team setup for your robotics team.
    - ii. Click on "+ Create new team"
    - iii. Enter the team name, eg; Admins
    - iv. Enter a description for the team, eg; Repository Admins
    - v. Select a parent team if this team should inherit permissions or members from another team. Typically, not used for Admins, but a Developers team might inherit the Admins team so the Admin members automatically become a member of the Developer team as well.
    - vi. Select the team visibility. Recommended to be Visible so others can identify team members responsible for a specific role.
    - vii. Push "Create team" when you are ready to create the new GitHub team.

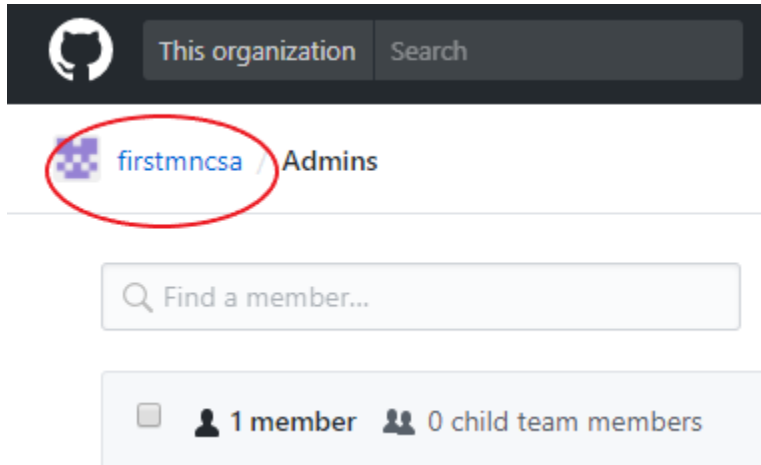
Create new team

A screenshot of the GitHub 'Create new team' form. The form has several sections: 'Team name' with a text input containing 'Admins' and a green checkmark; 'Mention this team in conversations as @firstmnscsa/admins.'; 'Description' with a text input containing 'Repository administrators'; 'Parent team' with a dropdown menu showing 'Select parent team'; 'Team visibility' with two radio button options: 'Visible' (selected, with 'Recommended' next to it) and 'Secret'. Below the 'Visible' option is a note: 'A visible team can be seen and @mentioned by every member of this organization.' Below the 'Secret' option is a note: 'A secret team can only be seen by its members.' At the bottom of the form is a green 'Create team' button, which is circled in red.

- viii. After creating the new team, you will be sent to the discussion page for your new team. Here you may start discussion topics others in your team will be able to see and reply to.
- ix. Next you may want to add individuals to the new team, you can do that by click on the “Members” tab next to “Discussions” and then click on “Add a member”



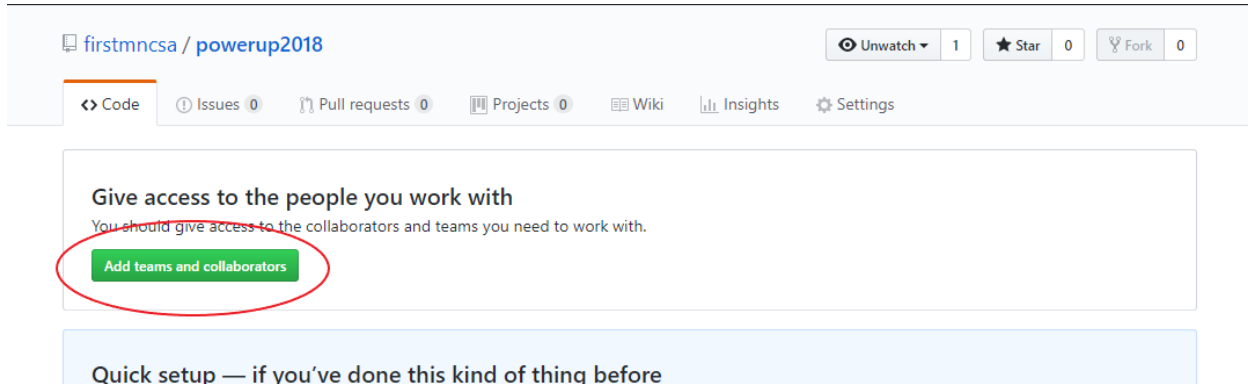
- x. Enter the username, full name or email address of the individual you want to add.
    - 1. Enter a GitHub username if the individual has a GitHub account already.
    - 2. Enter a Full Name if the individual has a GitHub account already.
    - 3. You can enter any email address regardless if it was used for a GitHub account already. If there's no existing GitHub account associated with the email address, they will be invited via email to join and create one.
  - xi. The remaining tabs, “Teams” “Repositories” and “Settings” most likely don't need to be changed at this point. But here's a short synopsis of what they are for.
    - 1. Teams – This is teams within teams, you can further organize members of this team into sub-teams. Eg; drivetrain developers, vision developers, etc...
    - 2. Repositories – This is where you can create repositories this team will own and maintain. You might want to use this to fork the teams central repository and then use that forks for team member commits. Or you may want to use this to store code specific for the purposes of the team or sub-team. Eg. Vision code.
    - 3. Settings – Any of the settings you used when creating the team can be changed here. Eg; Team name, visibility. This is also where you can delete the team if desired. Just be ware that if you have a repository associated with the team, it will be deleted as well.
15. Now that we have our team(s) created, we need to assign them to the repository.
- a. Navigate back to the “Settings” for the repository by click on the name of the organization or GitHub account in the upper left.



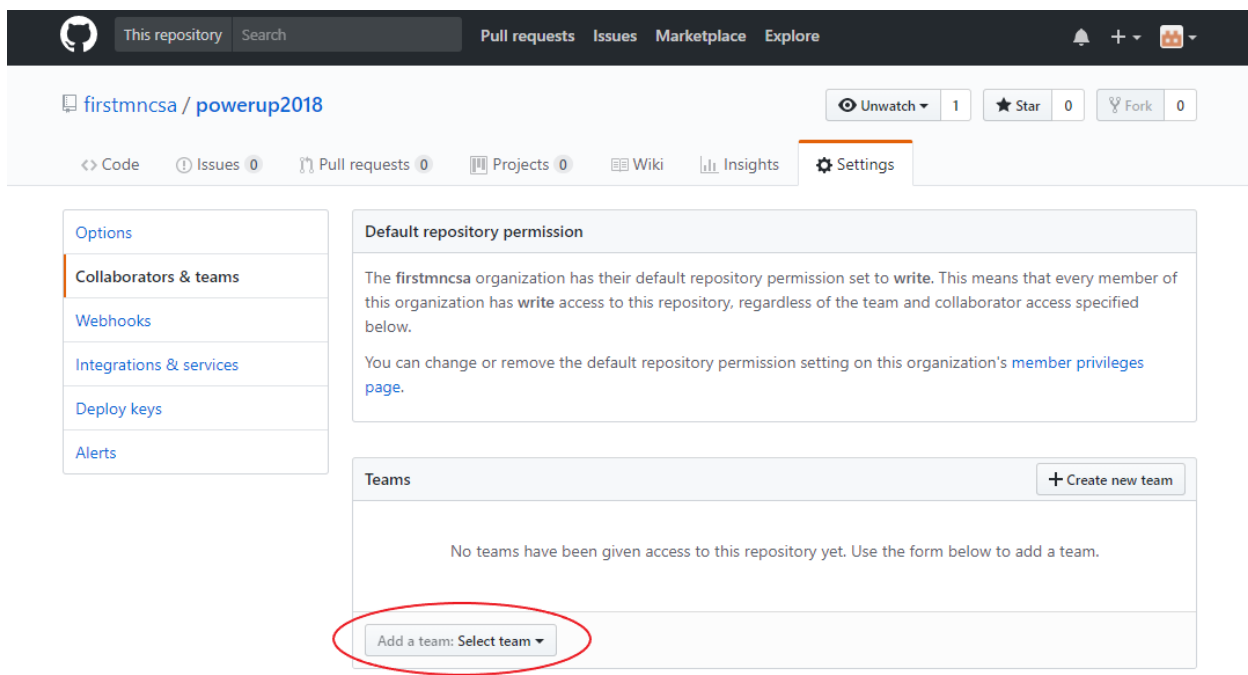
- b. Next click on “Repositories” tab and then click on the desired repository to add the team(s) to.



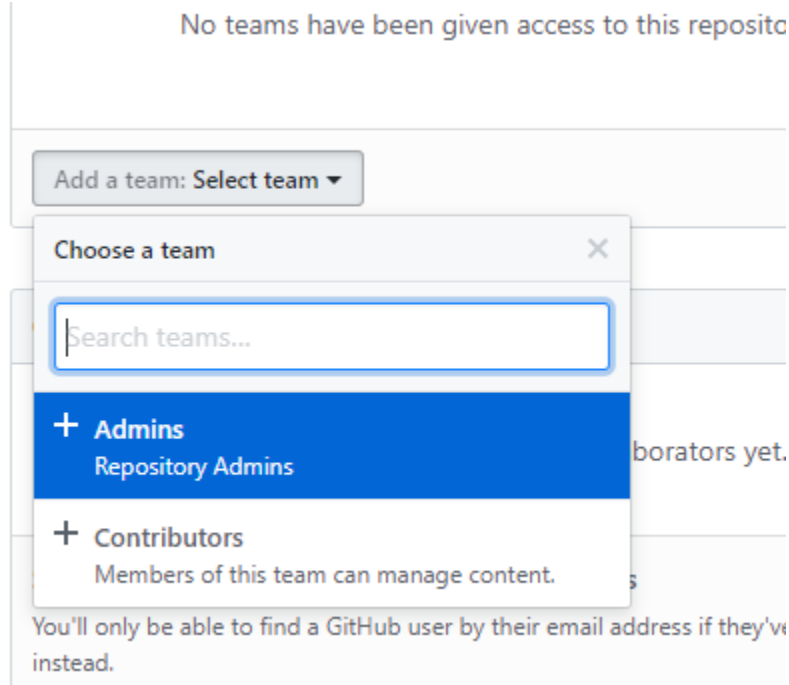
- c. Next click on “Add Teams” again or click on the “Settings” tab and then “Collaborators & teams”



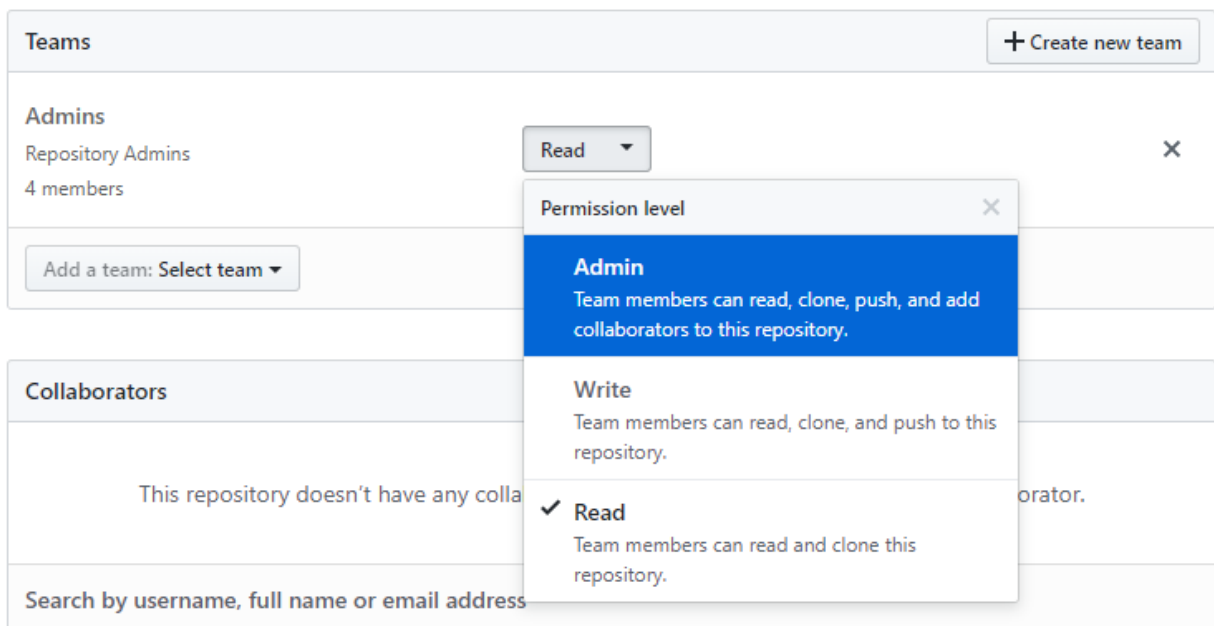
- d. Next, we need to add the team to this repository. Due this by selecting the team(s) from the “Select team” pull down menu.



- e. Select the team you want to add.



f. Next select the level of permissions the team should have.



g. Repeat the team selection and permission assignment for each team.



Year Round CSA Contact info: [firstmn.csa@gmail.com](mailto:firstmn.csa@gmail.com) Slack: [firstmn.csa.slack.com](https://firstmn.csa.slack.com)

First MN Website: <http://mnfirst.org/first-community-resources/local-assistance/>

First MN CSA Github: <https://github.com/firstmn.csa>

First MN Google Drive: <http://goo.gl/STtiAg>