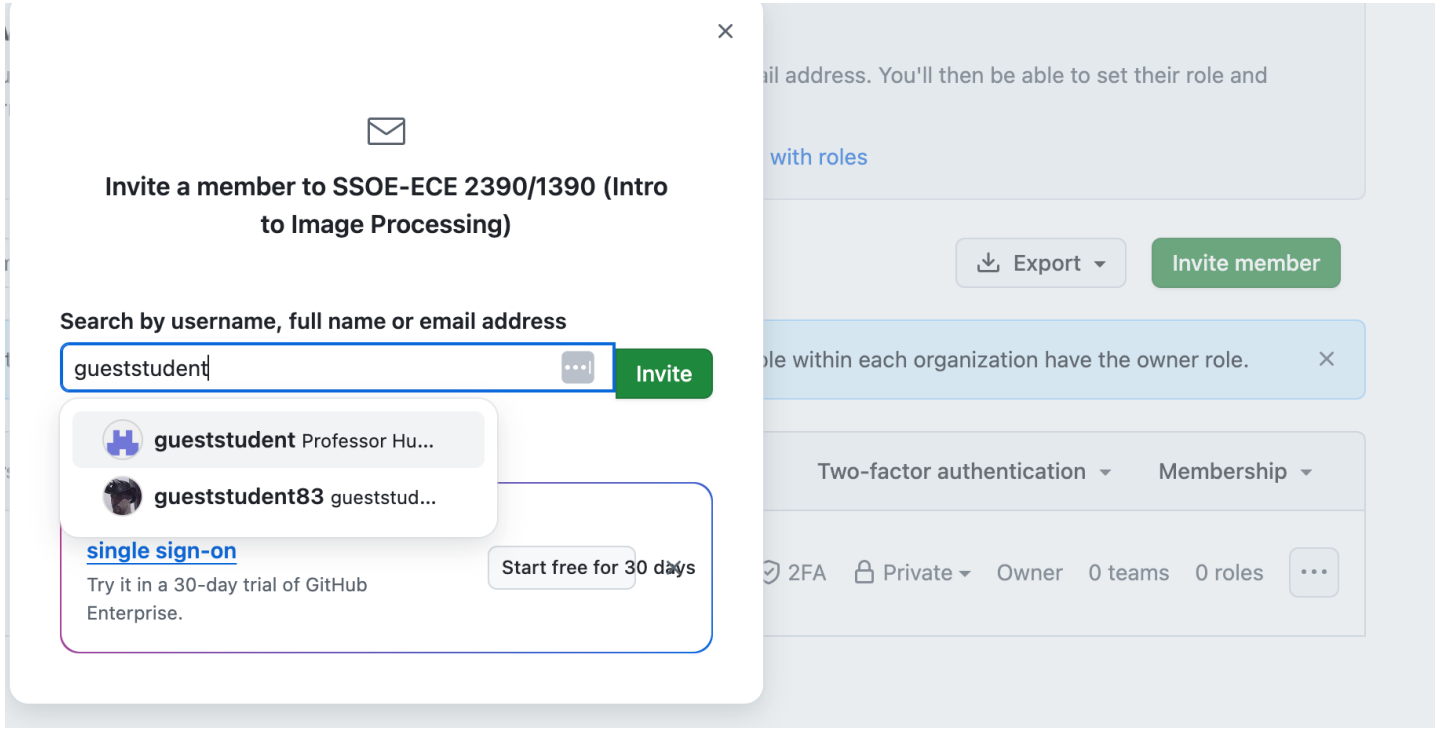
Introduction to GitHub Teams/Organizations/Classroom

Git Organization for course

- https://github.com/SSOE-ECE1390



Companies search your GitHub !

- Chose your GitHub username appropriately (and professionally)
- Private vs Public Repos

Homework 0

- Install VS-Code
  - https://code.visualstudio.com/
  - Add extensions for
    - GitHub Pull Requests (the official GitHub one)
    - GitHub Classroom
    - Python Language Support

Homework 0

- Python 3.12 (not 3.13)
  - Create dedicated virtual env
  - install:
    - opencv-python
    - matplotlib
    - numpy
    - mediapipe

Homework 0

- Clone the assignment repo (link found in canvas)

How this works with the org:

https://github.com/orgs/SSOE-ECE1390

  - Creates a fork of the template repo (assignment) with your Git name. E.g. homework0_huppertt
  - Make edits, commit, and push your assignment to GitHub
    - Either within VS-code or command line, etc
    - If you want to change/update your answer, send a new commit to GitHub.
    - We will grade based on what is on GitHub and (when appropriate) make comments/grading as push requests into your repo

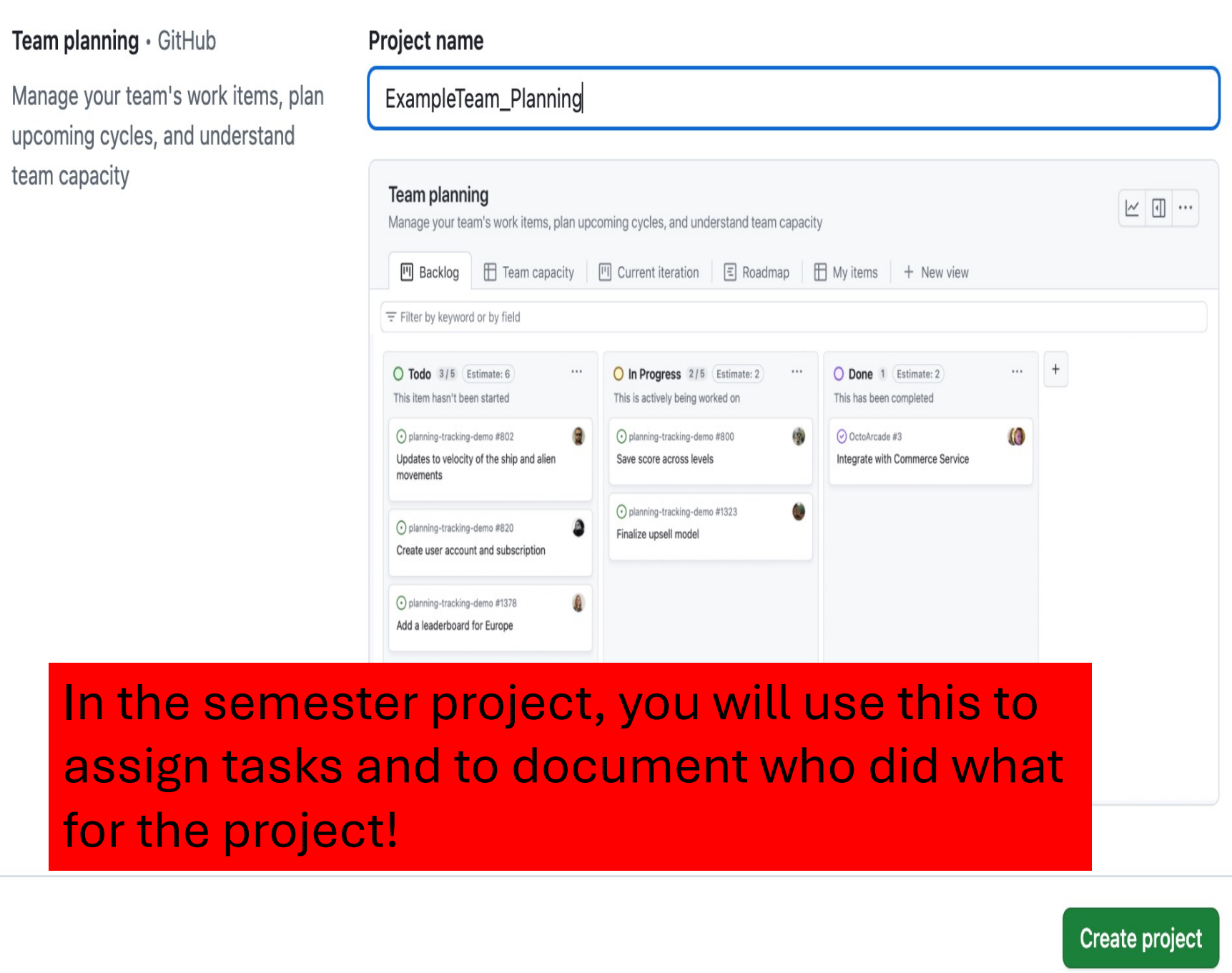
Group Project

https://github.com/SSOE-ECE1390

- Once you accept the first homework assignment and link it to your GitHub account, I can add you to the organization group.

- Once you have a team for your project, then one person from will create a new Team within the organization. Then add the other members of the team

- Create a Project in your team
- Create a Repo in your team



In the semester project, you will use this to assign tasks and to document who did what for the project!

GitHub Projects

- Not actual code
- Milestones/tasks /roadmaps
- Might refer to management of multiple repos as part of a larger project

Project settings
Manage access

Custom fields + New Field

Status
Sub-issues progress
Priority
Size
Estimate
Iteration
Start date
End date

Project settings

Project name

ExampleTeam

Short description

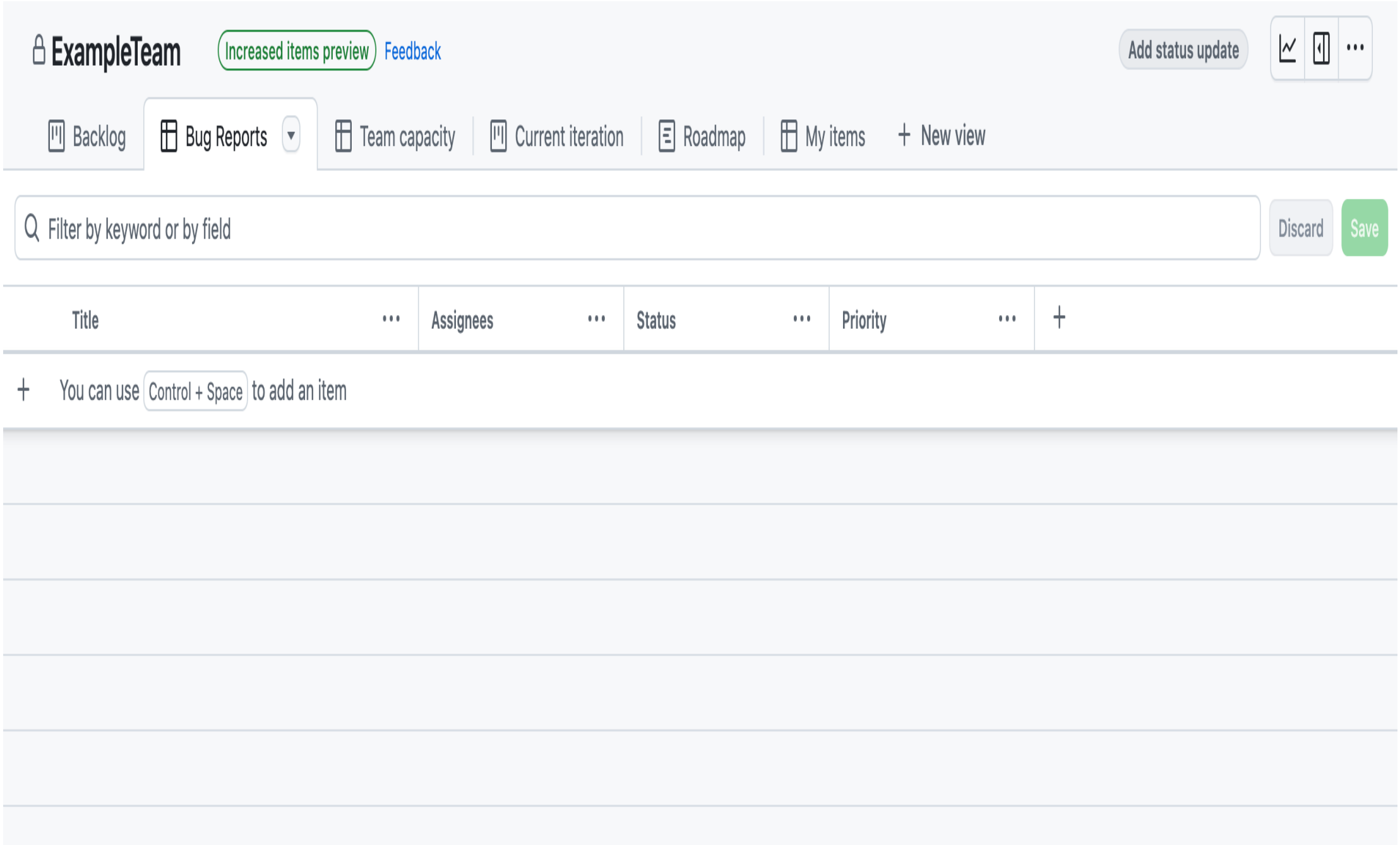
Virtual WebCam Project for ECE1390/2390

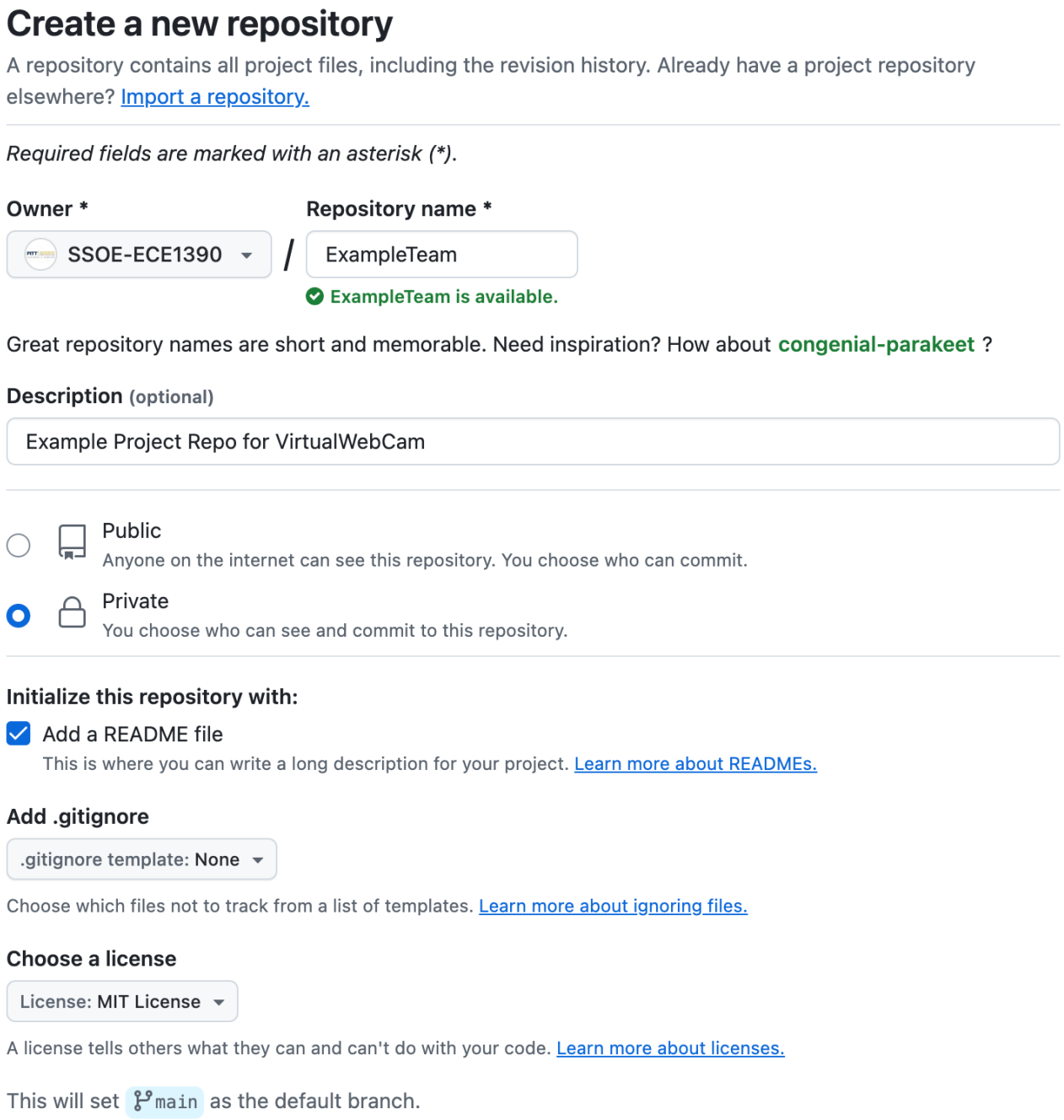
README

Write Preview

Virtual WebCam Project

A virtual webcam is a software program that re-broadcasts a video feed allowing modifications of the video. This project will build a virtual webcam to allow modification of a live camera feed into Zoom, which will allow the user to add visual effects onto the video.


ExampleTeam
Increased items preview
Feedback
Add status update
Backlog
Bug Reports
Team capacity
Current iteration
Roadmap
My items
New view
Filter by keyword or by field
Discard
Save
Title
Assignees
Status
Priority
You can use Control + Space to add an item



GitHub Repos

- Is the actual code
- You will have one main repo for your project; however, you will create branches from this repo such that each group member can work on their part of the code.
- I would like to see push requests and code review (for ECE 2390 groups, this is a requirement).

Creating a fork, making edits, and requesting review for a pull request

1) In the GitHub ORG site on your team, select the parent repo and create a fork
   → This creates a copy in YOUR GitHub account that only you have access to

2) Clone this repo from your account and make any edits. Commit and push the changes to your account.

3) When done, on the repo on your account, open a pull request. Add details about the changes. Select a reviewer for the request.

4) The reviewer will then get a notice that a pull request has been created and is ready for review. They can make comments and send back or approve and merge the request.

guststudent requested your review on this pull request. Add your review

Added details to the readme page for the project #2

Edit Code

Open **guststudent** wants to merge 1 commit into `main` from `1-add-readme-about-project`

Conversation 0 | Commits 1 | Checks 0 | Files changed 1 +20 −0

guststudent commented 2 minutes ago

Added the readme file to the project as part of the first homework assignment in the class.

Added details to the readme page for the project 38f7608

guststudent added the documentation label 2 minutes ago

guststudent requested a review from **huppertt** 2 minutes ago

Reviewers

huppertt

Still in progress? Learn about draft PRs

Assignees

No one—assign yourself

Labels

documentation

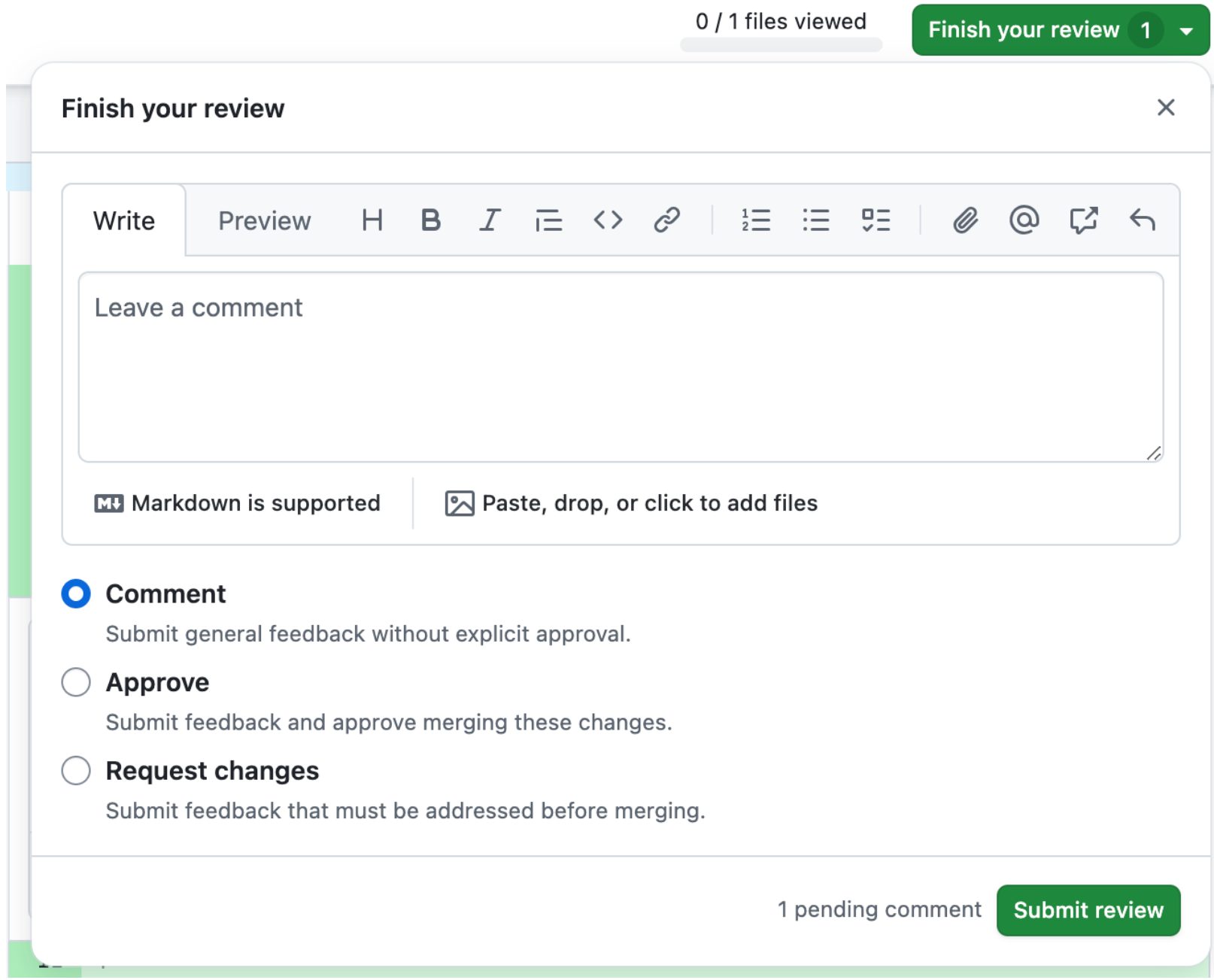
```
1   # ExampleTeam
2   Example Project Repo for VirtualWebCam
3  +
4  + Brief Description:
5  + The goal of this project is to build a virtual webcam to allow manipulations of
     live-video feeds from a webcam to be used in Zoom
6  +
7  + Team Members:
8  + Dr Huppert (huppert1@pitt.edu)
9  + The class TA (someone@pitt.edu)
10 +
```

huppertt Pending

Add GuestStudent@pitt.edu

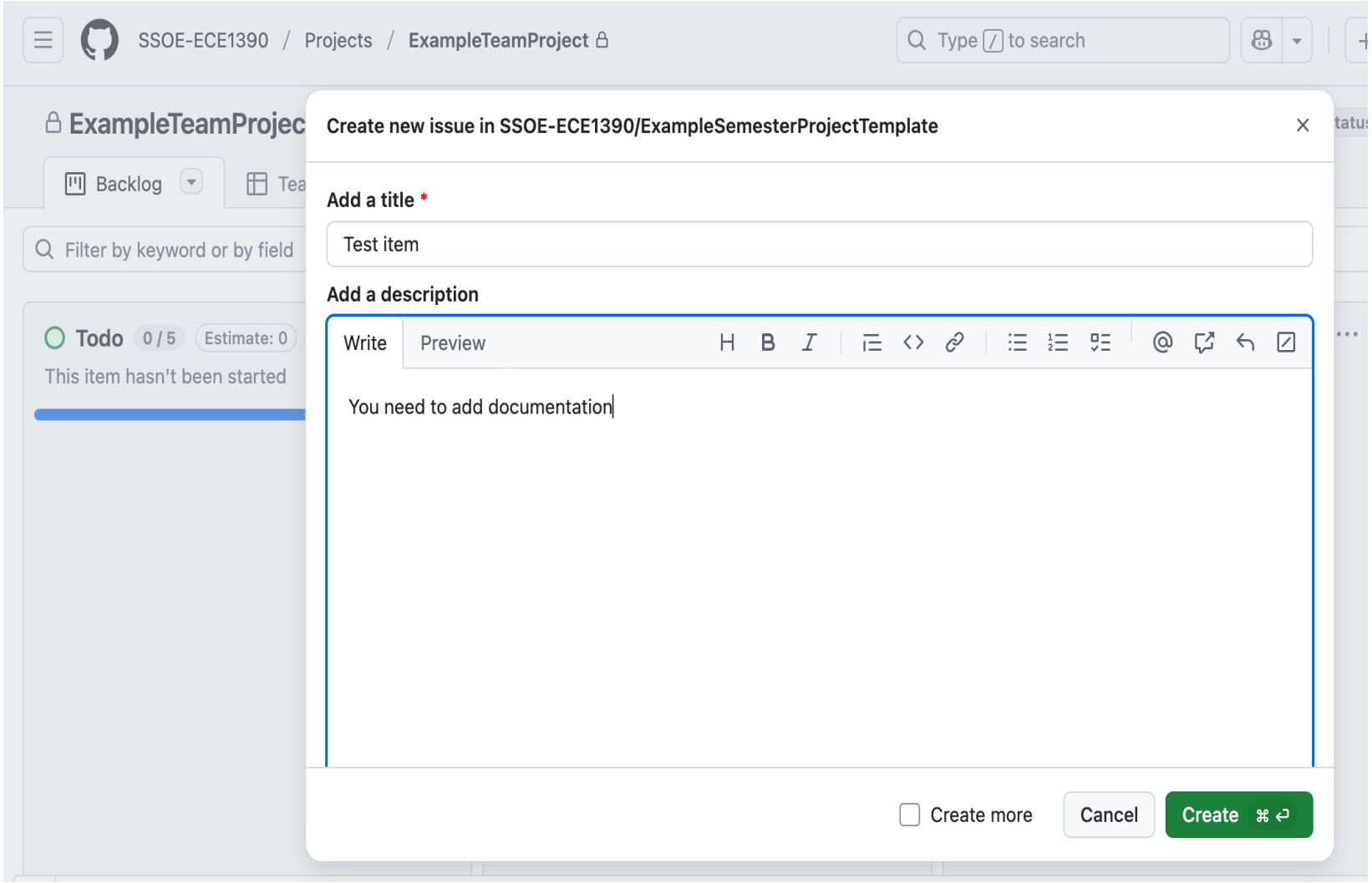
Reply...

```
11 +
12 + Proposed project specs:
13 + * Background replacement (greenscreen)
14 +     Include feature to change background of video
```


0 / 1 files viewed
Finish your review 1
Finish your review
Write
Preview
Leave a comment
Markdown is supported
Paste, drop, or click to add files
Comment
Submit general feedback without explicit approval.
Approve
Submit feedback and approve merging these changes.
Request changes
Submit feedback that must be addressed before merging.
1 pending comment
Submit review

Acting on issues

In the Project, create an issue by adding it to the workflow



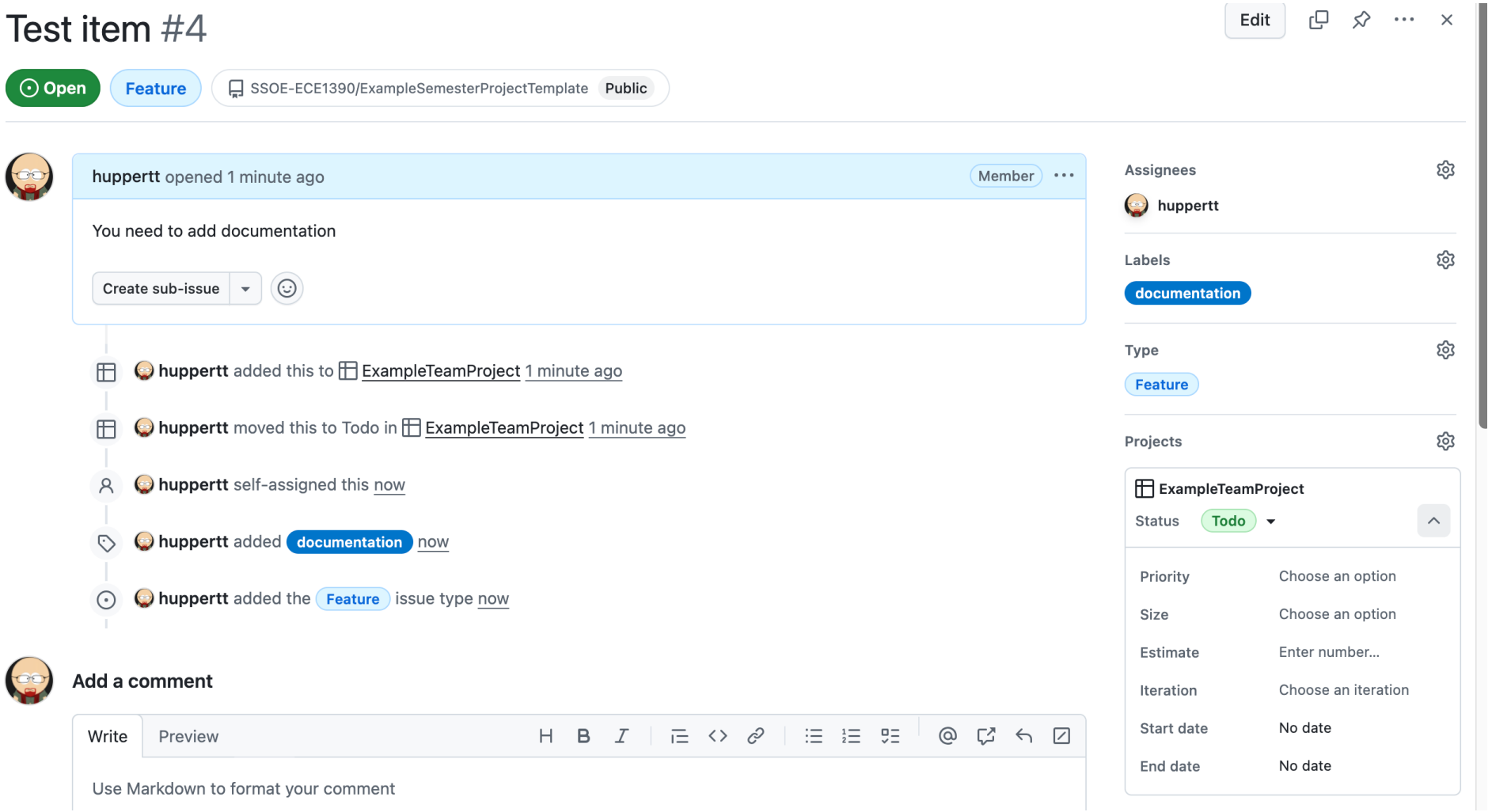
Acting on issues

Once created, you can open the issue and add more details

Assign to a person on the team

Add info about what type of issue this is

Add comments or attach files

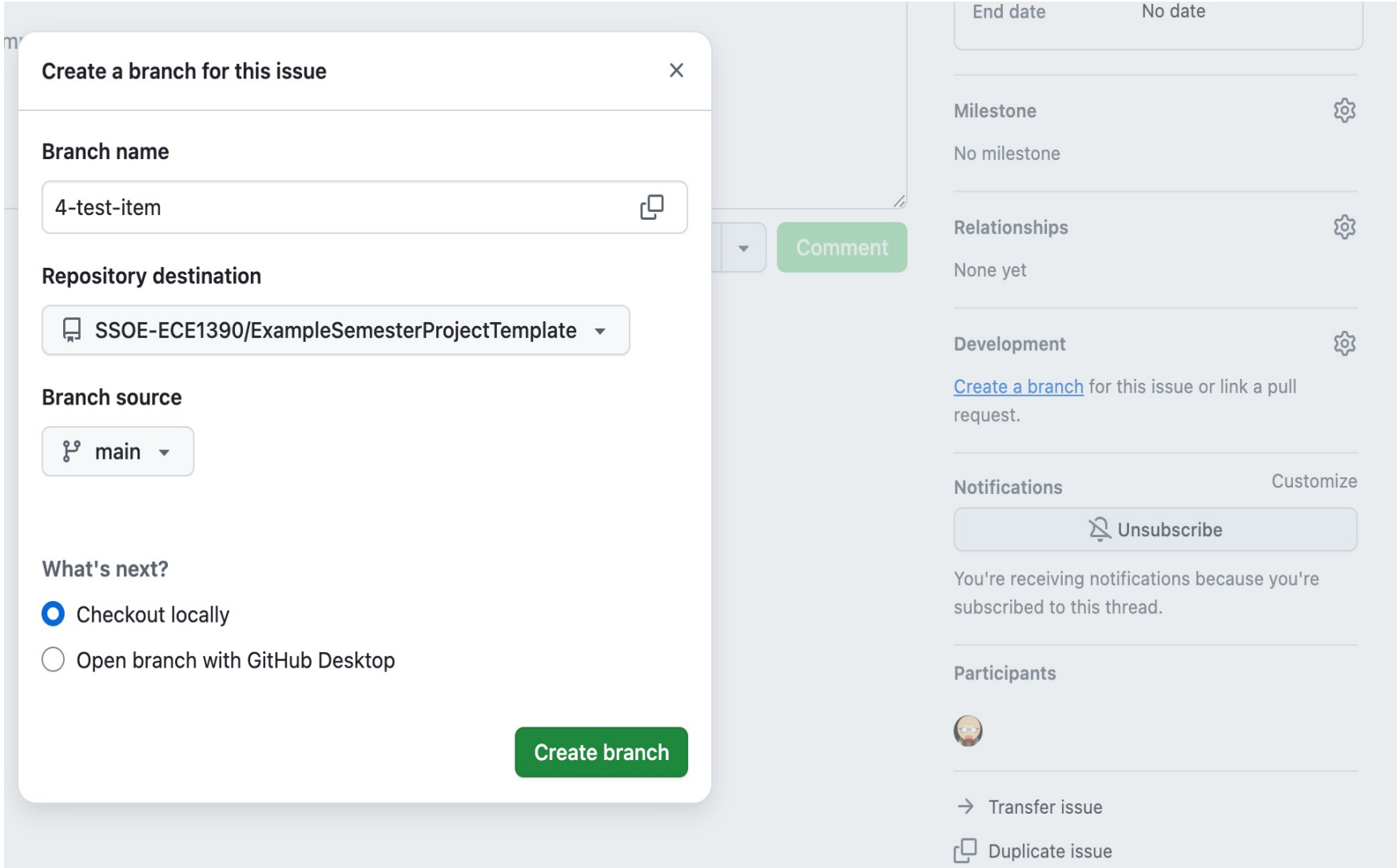


Acting on issues

At the bottom, you can create a branch or assign it to an existing pull request.

Branches – stay with the Org account. These are sub-repos

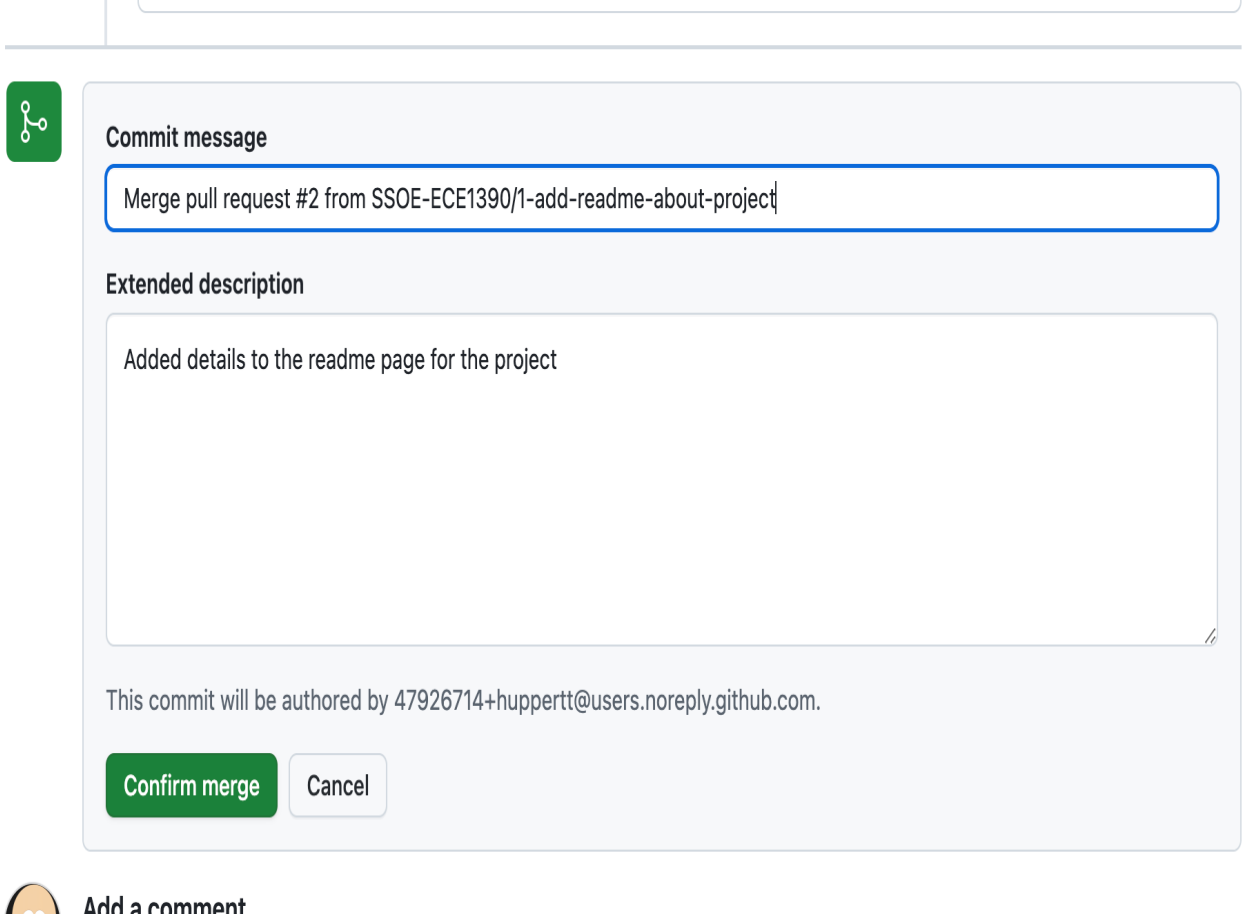
Forks – your local GitHub. These are copies of the repo.



Acting on issues

If you are working with branches, you need to switch the branch you are working on, make edits, and then eventually merge with a pull request.

If you are working with forks, you need to clone the fork, make edits, and then eventually merge with a pull request.



Acting on issues

Once an issue is completed, move it to the DONE category.

If you close an issue, it disappears (which is fine if you aren't tracking issues), but companies often want to save documentation of such changes.