How to use GitLab

To enable cross-functional and diverse teams

About me

Nuritzi Sanchez

Sr. Open Source Program Manager at GitLab

- Joined GitLab in February 2020
- Background in: program/project management, product management, business operations
- Have been part of Engagement team and Board of Directors at GNOME, LAS org, CHAOSS, and FOSS Responders

... I am not an engineer.
I am a leader, and an organizer, and I manage, and work with, cross-functional teams. I use GitLab in my everyday work

What we'll cover today

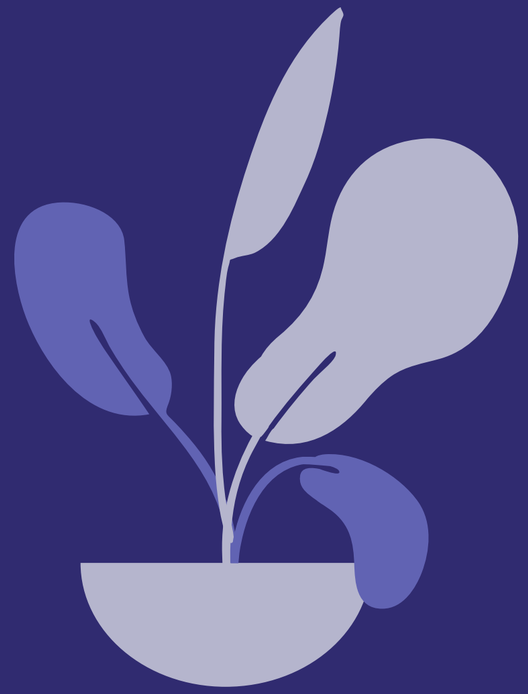
Why use GitLab cross-functionally

Creating the right environment

Tips & tricks for non-engineers

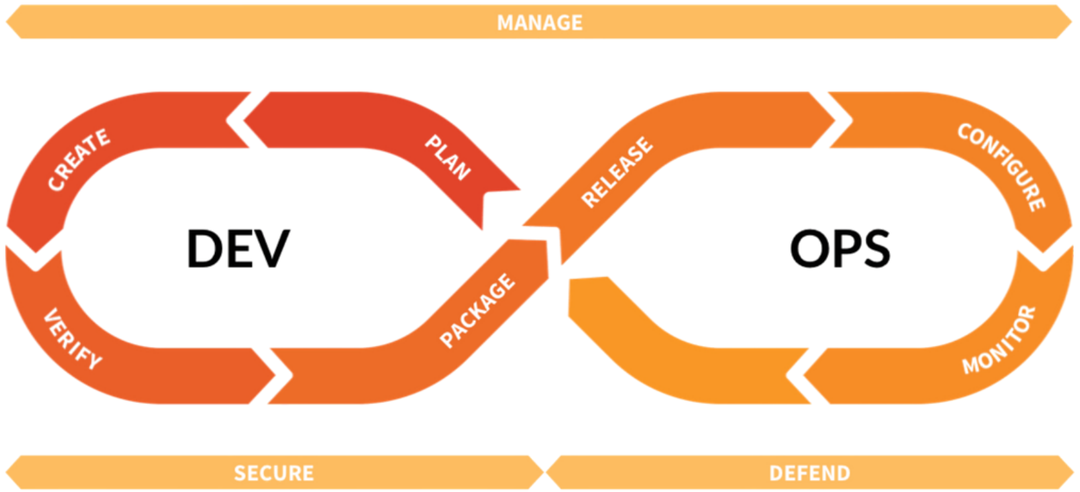
Why it matters

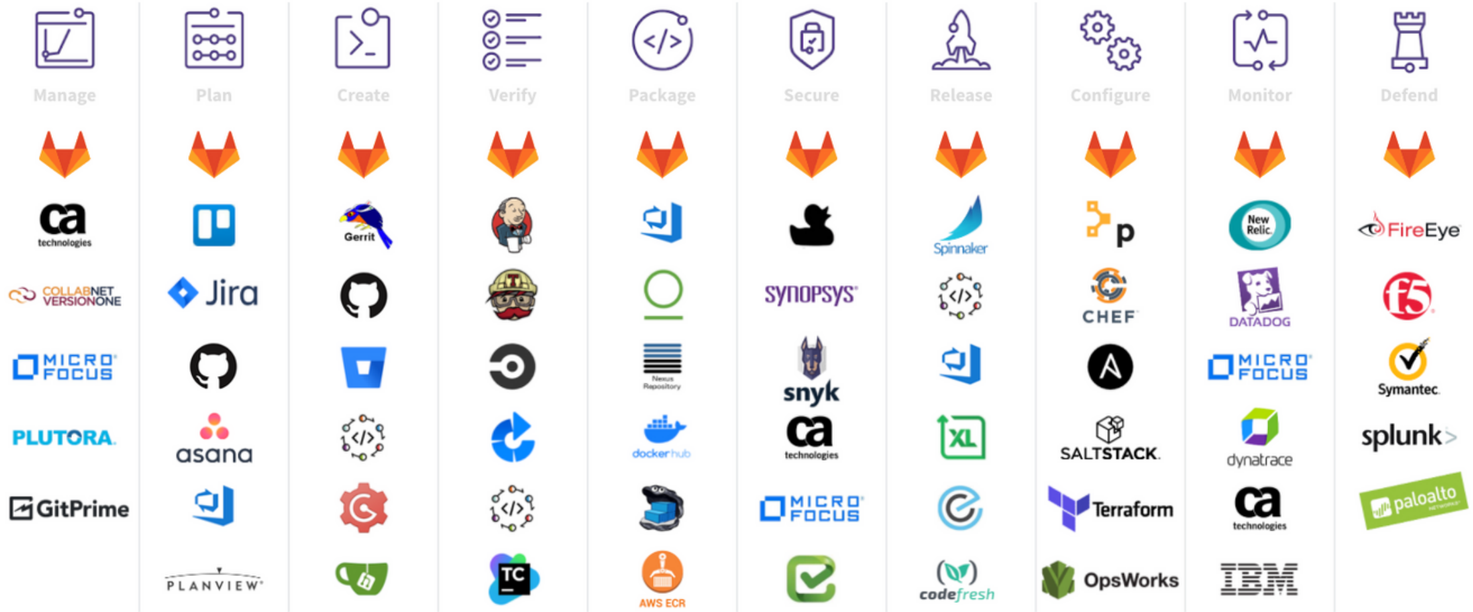
Reasons to use GitLab cross-functionally



One platform for the whole team

- Manage
- Plan
- Create
- Verify
- Package
- Secure
- Release
- Configure
- Defend





We need a broader set of skills and diversity in open source

Let's think beyond what's currently counted as a non-code contribution and where that contribution may come from

Sales & Business Development

Someone who is interested in gaining or using these skills may be interested in...

HR & People

Someone who is interested in gaining or using these skills may be interested in...

OSS: Fundraising, partnerships

Both of these things require you to pitch the value of the open source community / project and require you to develop your communication and negotiation skills, among other things

OSS: Engagement, Board of Directors, newcomers initiatives

We need people who want to make the community awesome. Opportunities include working on or starting newcomers initiatives and lowering the barrier of entry, as well as retention. Can also help with community ommunication practices.

Using one platform allows for

- Transparency into other teams’ work
- Easier communication
- Ability to contribute to other teams

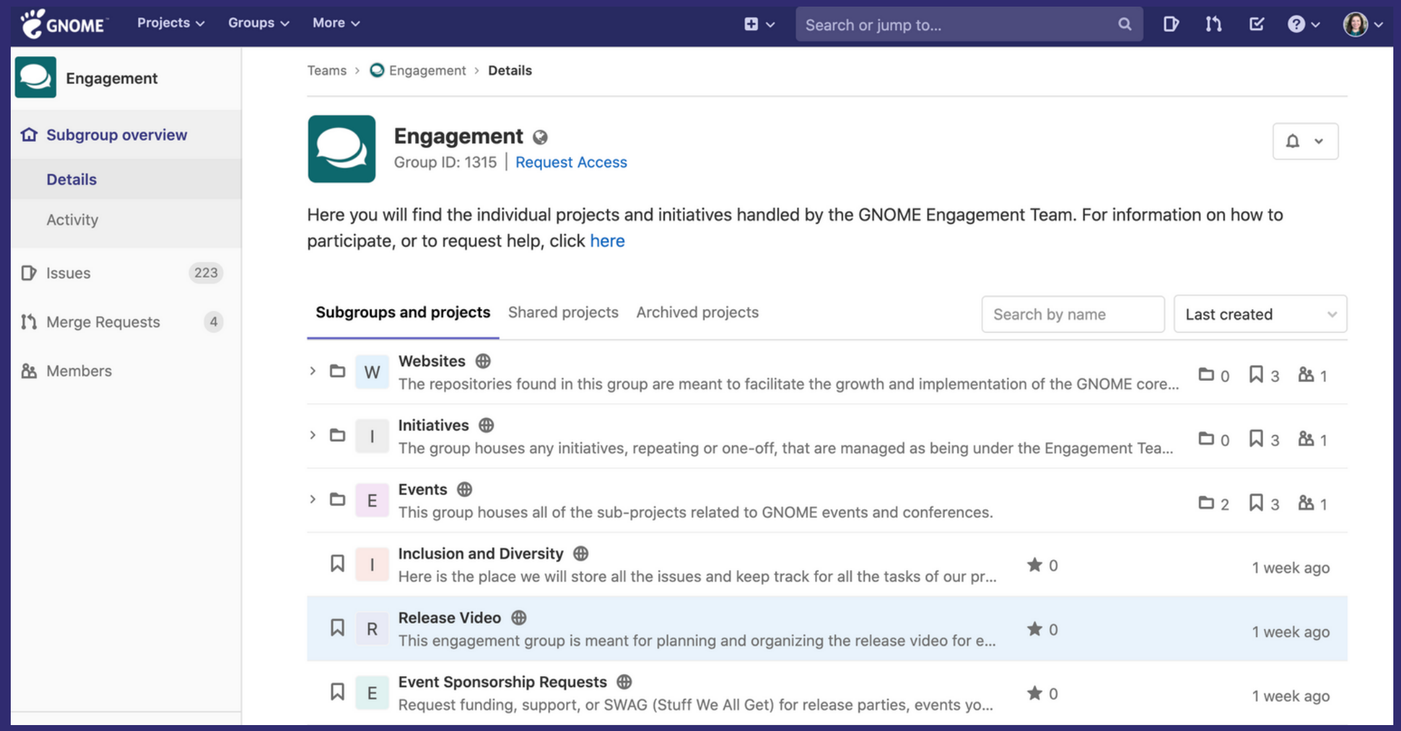
Transparency

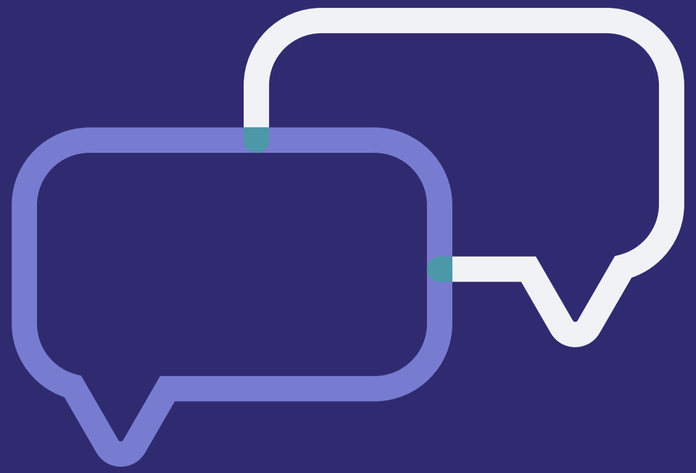
Example

The Engagement team uses GitLab.

This means engineers, docs team, designers, and others can pop over any time and help with any items that involve their team.

Our community can benefit from expert involvement.





Easier communication

Fragmentation among tools can hinder communication.

When everyone is on the same tool, then you're able to collaborate more efficiently.

Ability to contribute to other teams

I started off by doing purely strategy + project management tasks for LAS.

Then, I started editing the LAS website's markdown pages, and adding new files.

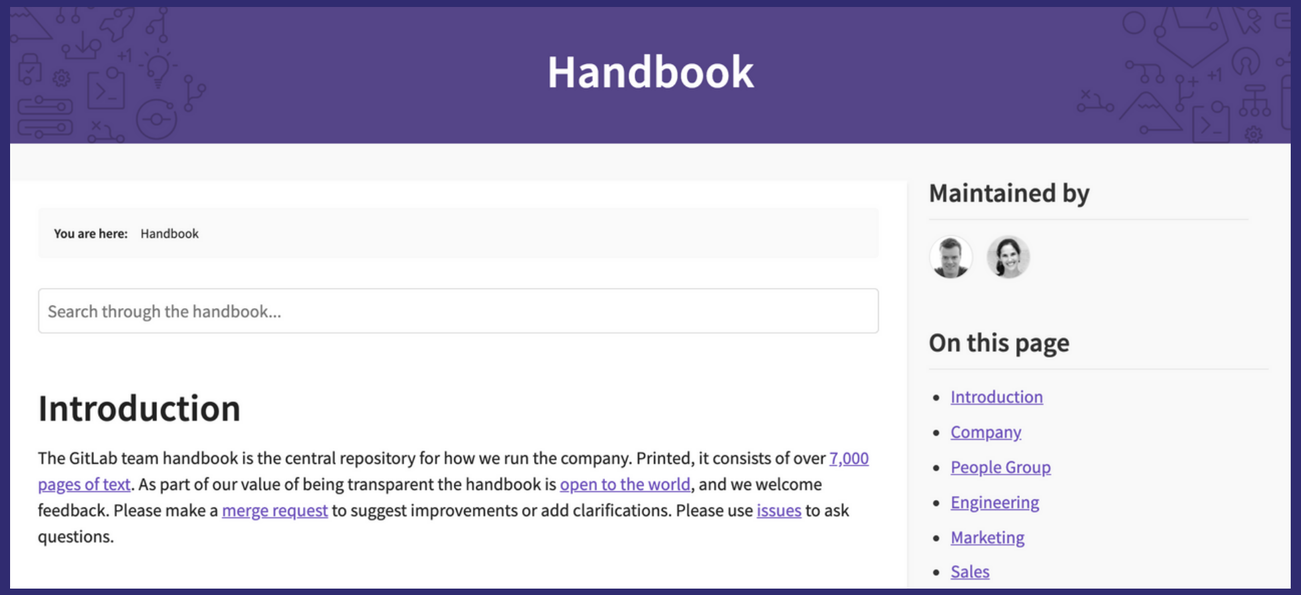
Then, I joined GitLab and had an easier time creating and reviewing MRs, and updating the GitLab website.

Now, I feel more comfortable looking at websites' code and making MRs!

Environment

Creating the right environment is key to enabling cross-functional team collaboration and fostering a diverse community

Documentation is key



At GitLab, we have a handbook-first approach and document all of our business practices.

It's integrated into our website, and "code owners" maintain each page.

Different teams have different processes. Docs help people navigate cross-team collaboration.

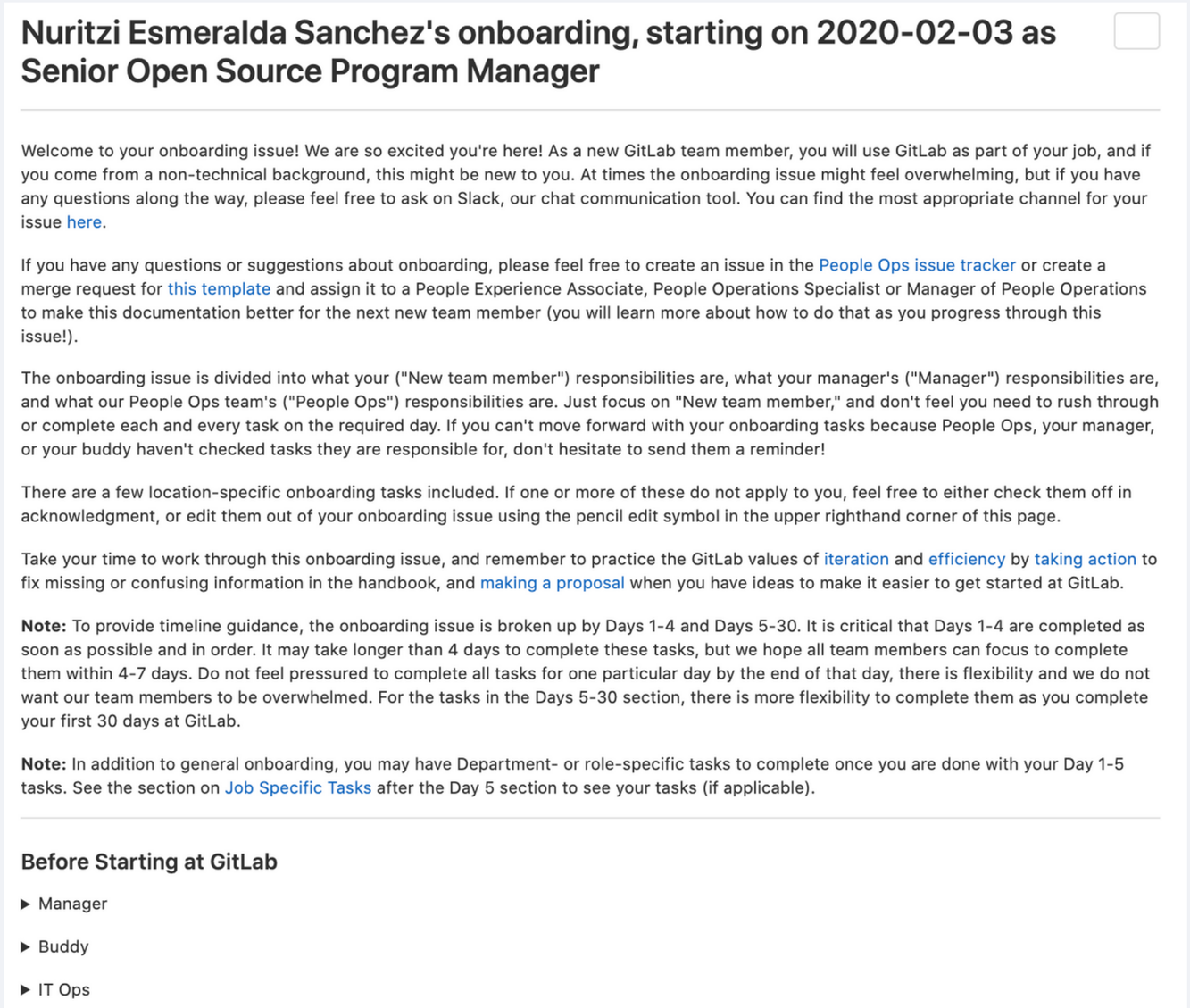
Documentation enables a remote culture that depends on asynchronous communication.

Onboarding and training

There are many types of onboarding, for example:

- Tools onboarding
- Subject matter onboarding
- Social onboarding

Creating an onboarding issue template can help people self-serve



Nuritzi Esmeralda Sanchez's onboarding, starting on 2020-02-03 as Senior Open Source Program Manager

Welcome to your onboarding issue! We are so excited you're here! As a new GitLab team member, you will use GitLab as part of your job, and if you come from a non-technical background, this might be new to you. At times the onboarding issue might feel overwhelming, but if you have any questions along the way, please feel free to ask on Slack, our chat communication tool. You can find the most appropriate channel for your issue here.

If you have any questions or suggestions about onboarding, please feel free to create an issue in the People Ops issue tracker or create a merge request for this template and assign it to a People Experience Associate, People Operations Specialist or Manager of People Operations to make this documentation better for the next new team member (you will learn more about how to do that as you progress through this issue!).

The onboarding issue is divided into what your ("New team member") responsibilities are, what your manager's ("Manager") responsibilities are, and what our People Ops team's ("People Ops") responsibilities are. Just focus on "New team member," and don't feel you need to rush through or complete each and every task on the required day. If you can't move forward with your onboarding tasks because People Ops, your manager, or your buddy haven't checked tasks they are responsible for, don't hesitate to send them a reminder!

There are a few location-specific onboarding tasks included. If one or more of these do not apply to you, feel free to either check them off in acknowledgment, or edit them out of your onboarding issue using the pencil edit symbol in the upper righthand corner of this page.

Take your time to work through this onboarding issue, and remember to practice the GitLab values of iteration and efficiency by taking action to fix missing or confusing information in the handbook, and making a proposal when you have ideas to make it easier to get started at GitLab.

Note: To provide timeline guidance, the onboarding issue is broken up by Days 1-4 and Days 5-30. It is critical that Days 1-4 are completed as soon as possible and in order. It may take longer than 4 days to complete these tasks, but we hope all team members can focus to complete them within 4-7 days. Do not feel pressured to complete all tasks for one particular day by the end of that day, there is flexibility and we do not want our team members to be overwhelmed. For the tasks in the Days 5-30 section, there is more flexibility to complete them as you complete your first 30 days at GitLab.

Note: In addition to general onboarding, you may have Department- or role-specific tasks to complete once you are done with your Day 1-5 tasks. See the section on Job Specific Tasks after the Day 5 section to see your tasks (if applicable).

Before Starting at GitLab

▶ Manager

▶ Buddy

▶ IT Ops

An onboarding issue template can be created with information like:

- History of the project
- Video recordings for how to use tools
- Tasks that all community members should do (read CoC, add to their profile page, etc)
- Instructions for scheduling coffee chats with team, or live onboarding sessions
- Links to information that you want all members to know
- Tasks to complete per team (Design, Engineering, Engagement)

Think about non-engineers regularly

This can influence how you do a lot of things.

Example:

The GitLab website has included links at the bottom where you can edit the page's markdown or open the page directly in WebIDE.

Edit this page — open Web IDE — please contribute.

CC BY-SA

Tips & Tricks

Some of my favorite GitLab tricks to help other non-engineers

Community Edition
&
Enterprise Edition

https://about.gitlab.com/solutions/open-source/

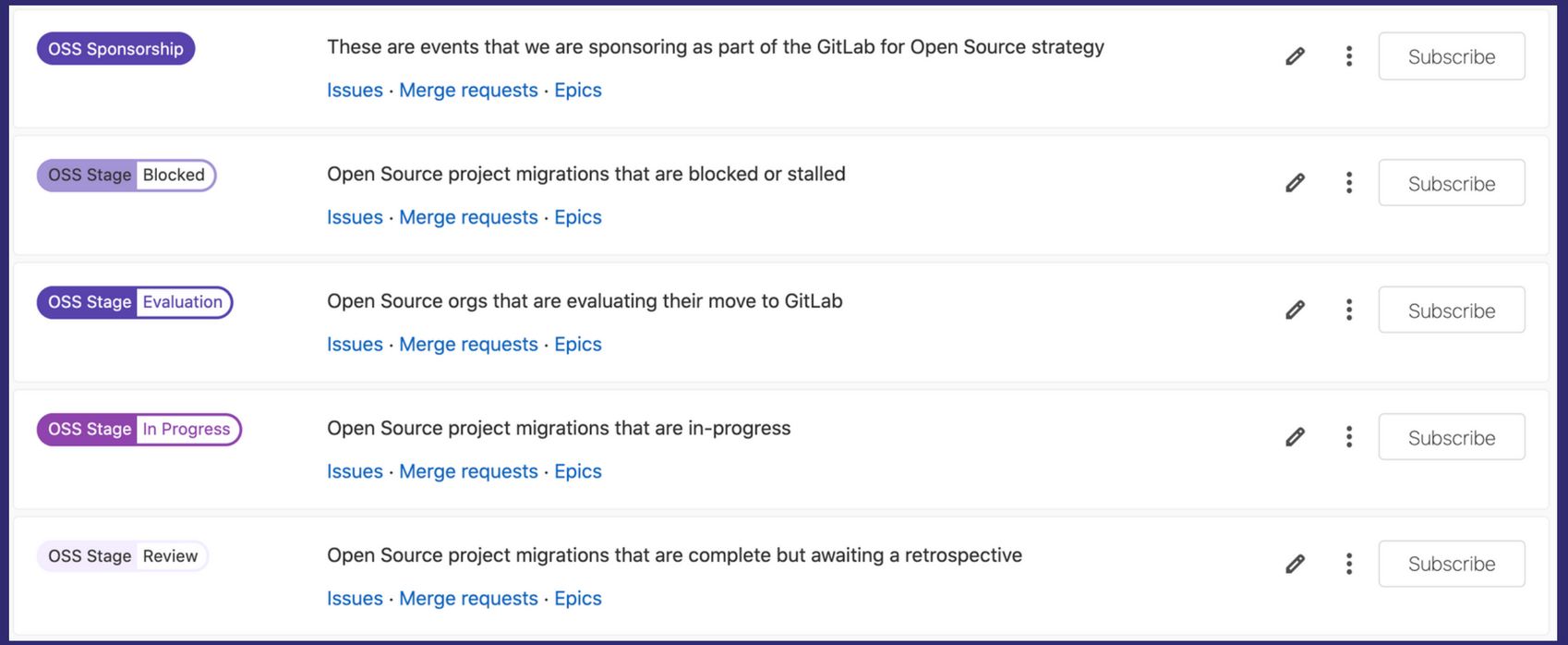
Labels

Community Edition:

- Labels are really flexible
- Subscribe to them

Enterprise Edition:

- Scoped labels

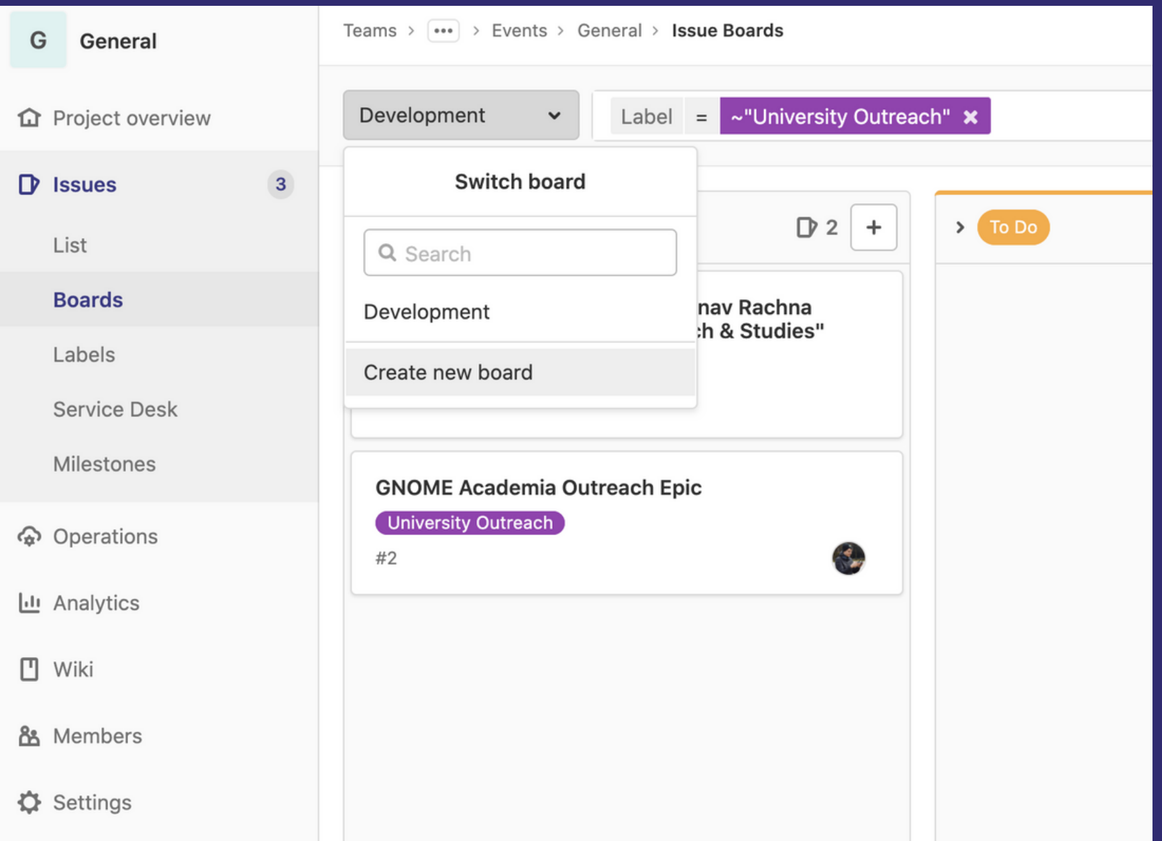


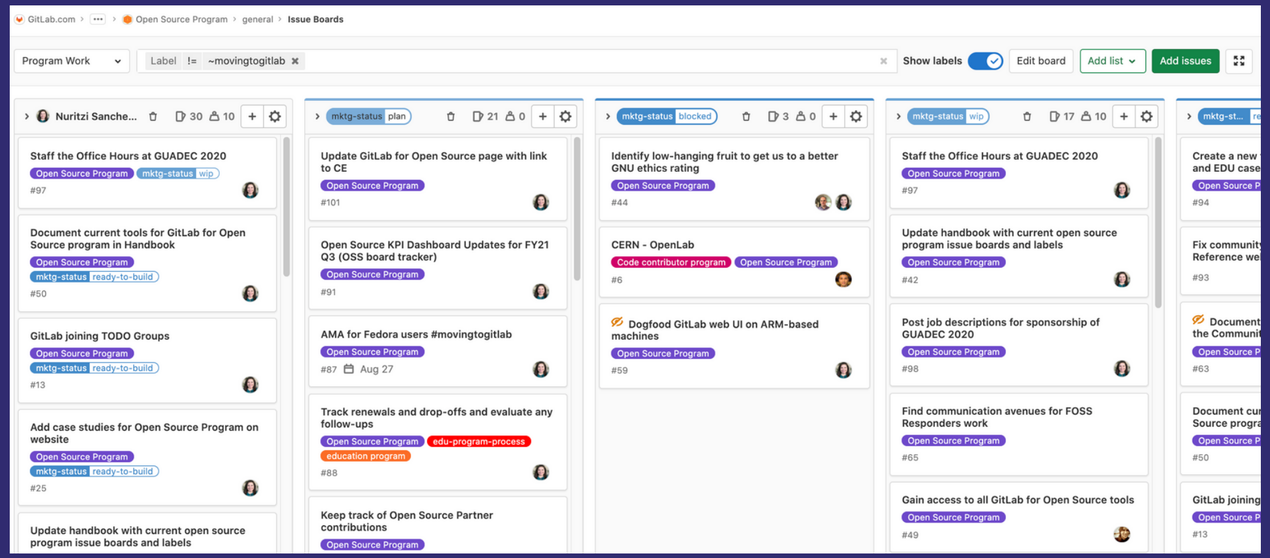
Kanban boards

Use them to track the status of issues or sort by type

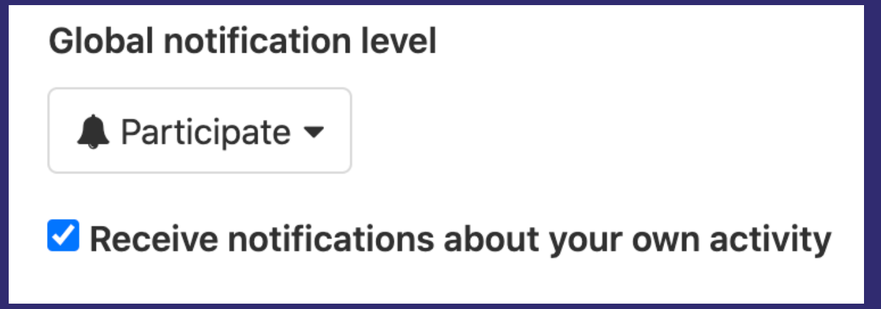
Create new ones with labels that you're tracking

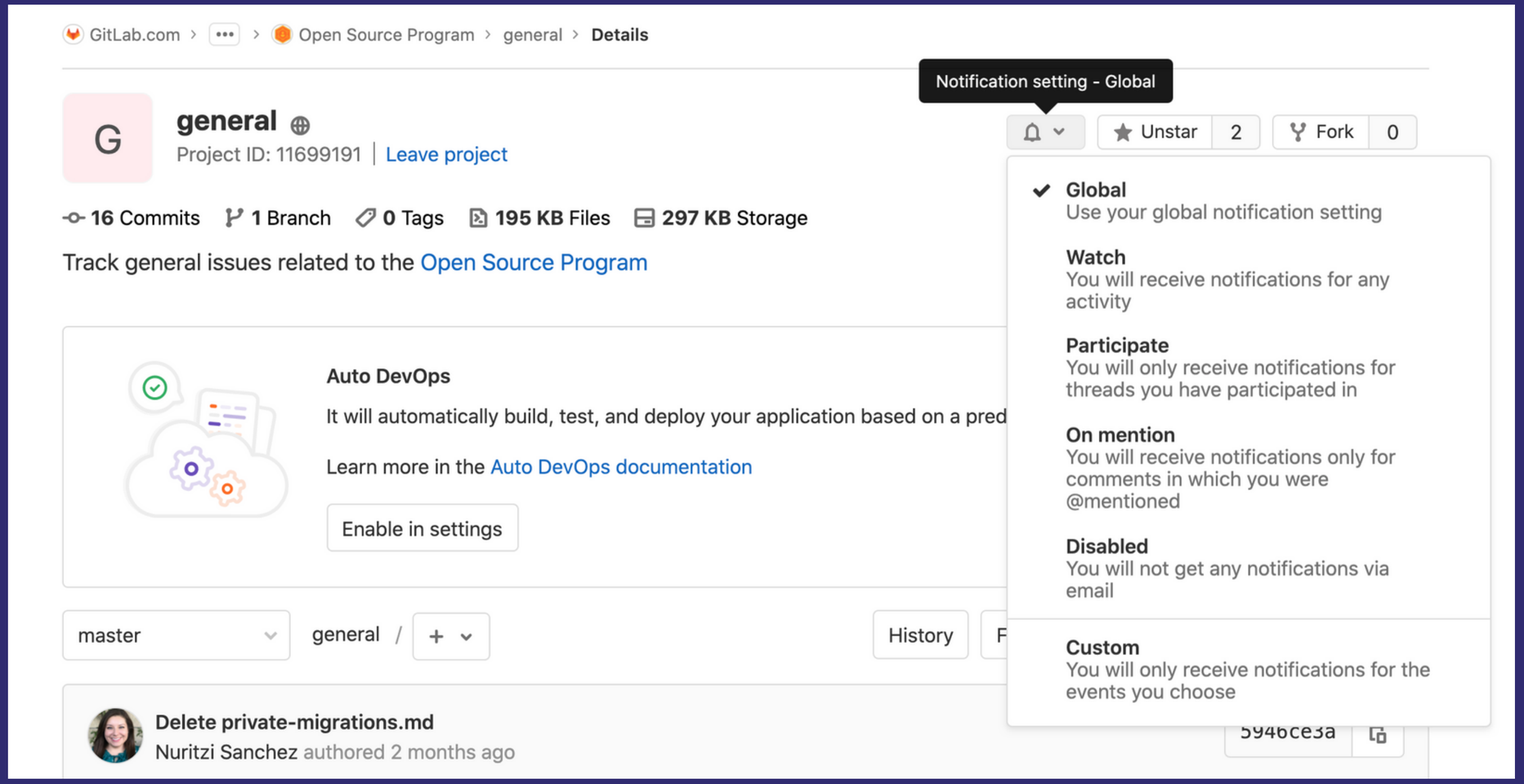
Can also be used to see who is assigned by issues in one column



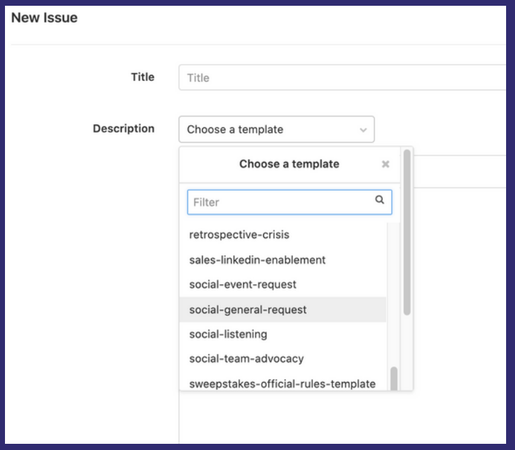


Notifications





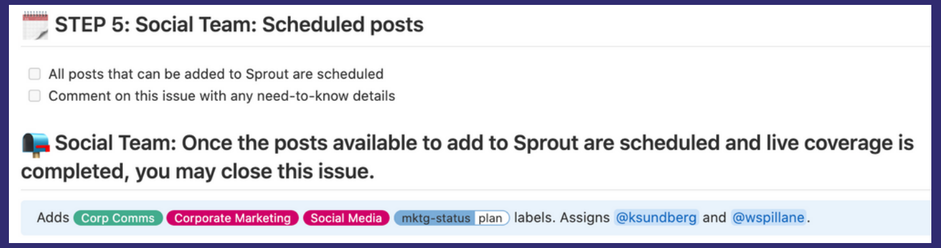
Issue Templates

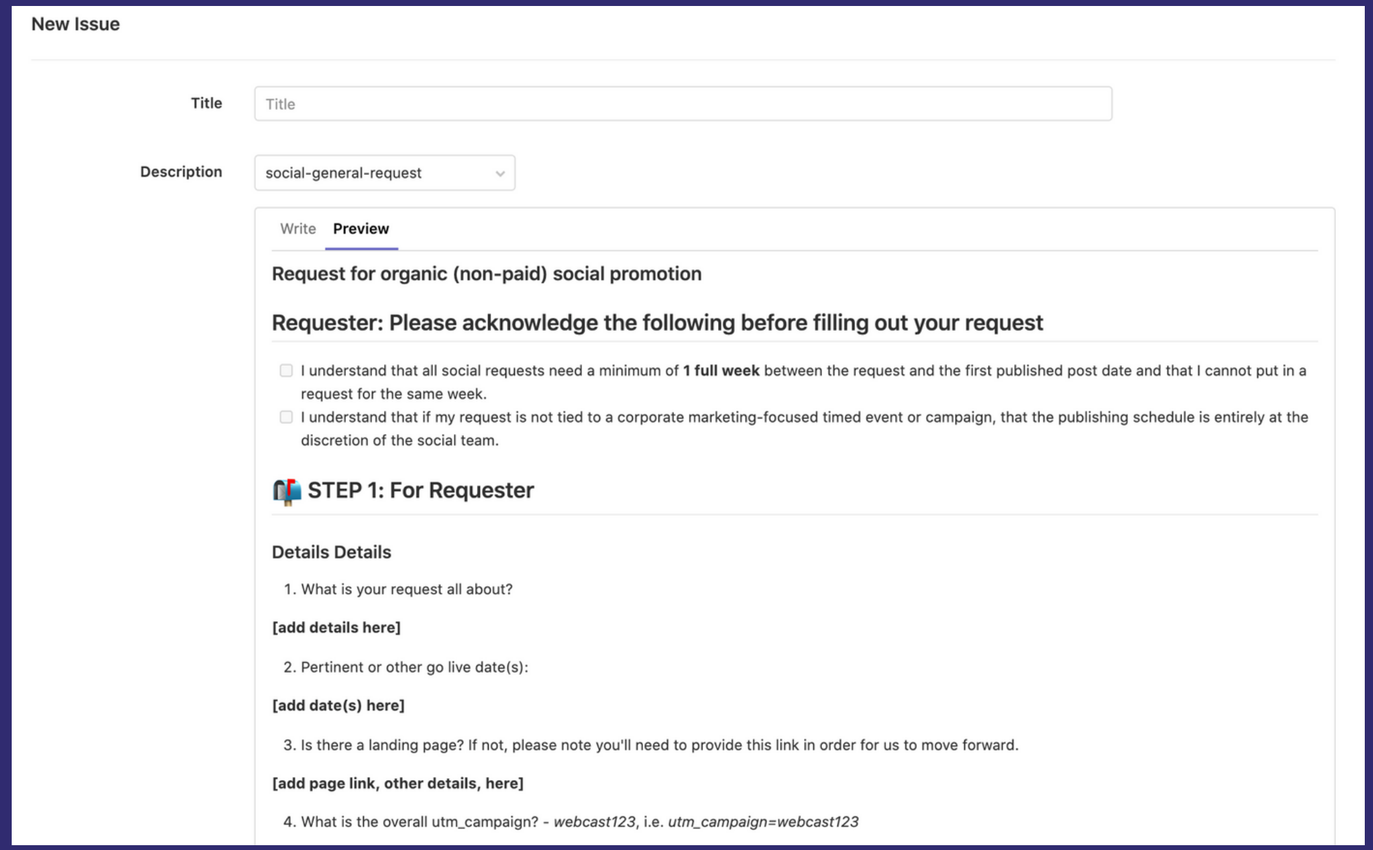


So important!

Train people how to create templates on their own. Include a link to documentation for this in your onboarding issue.

Add automatic labels and tagging of users

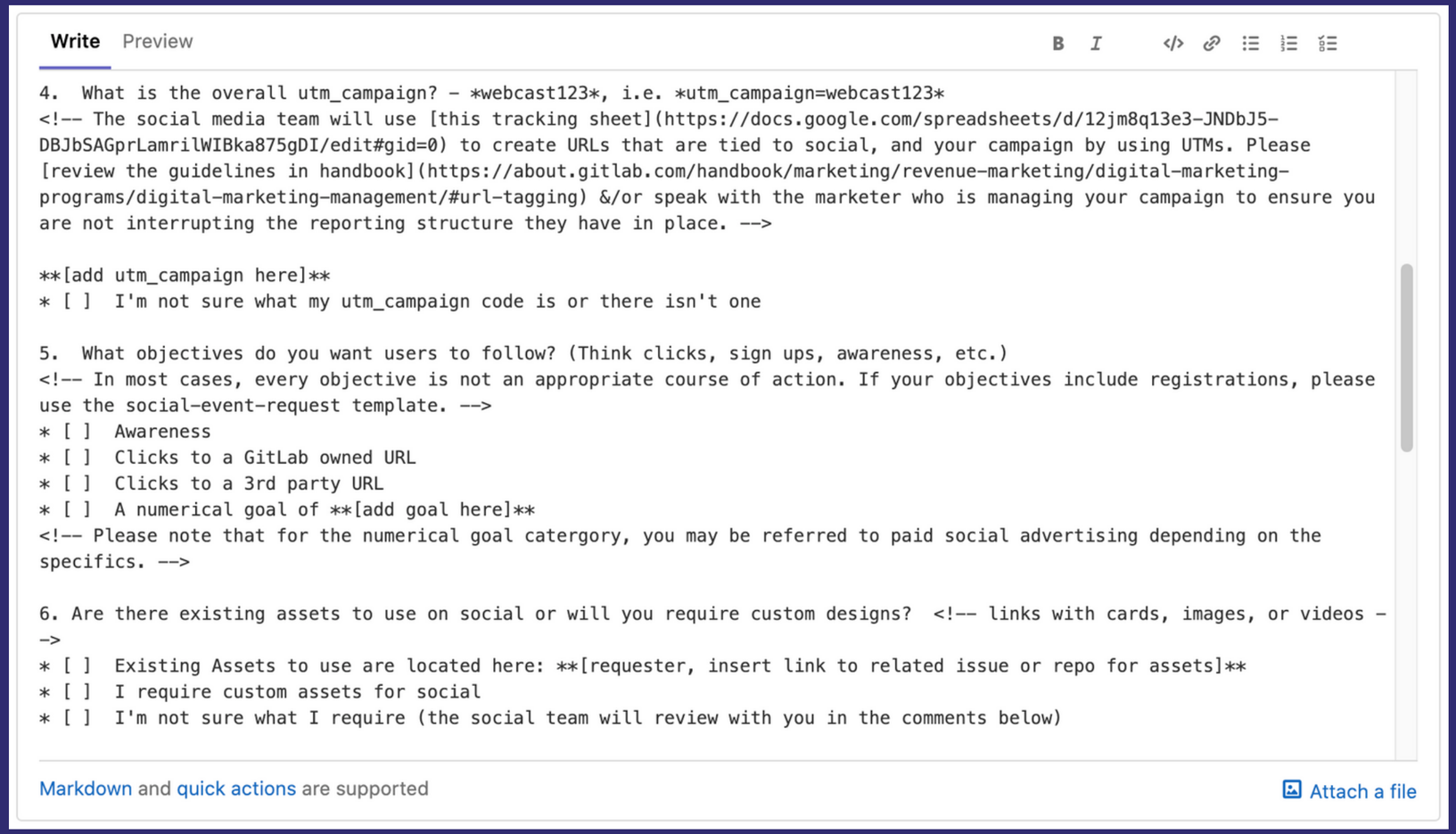




https://gitlab.com/gitlab-com/marketing/corporate_marketing/corporate-marketing/-/issues/new

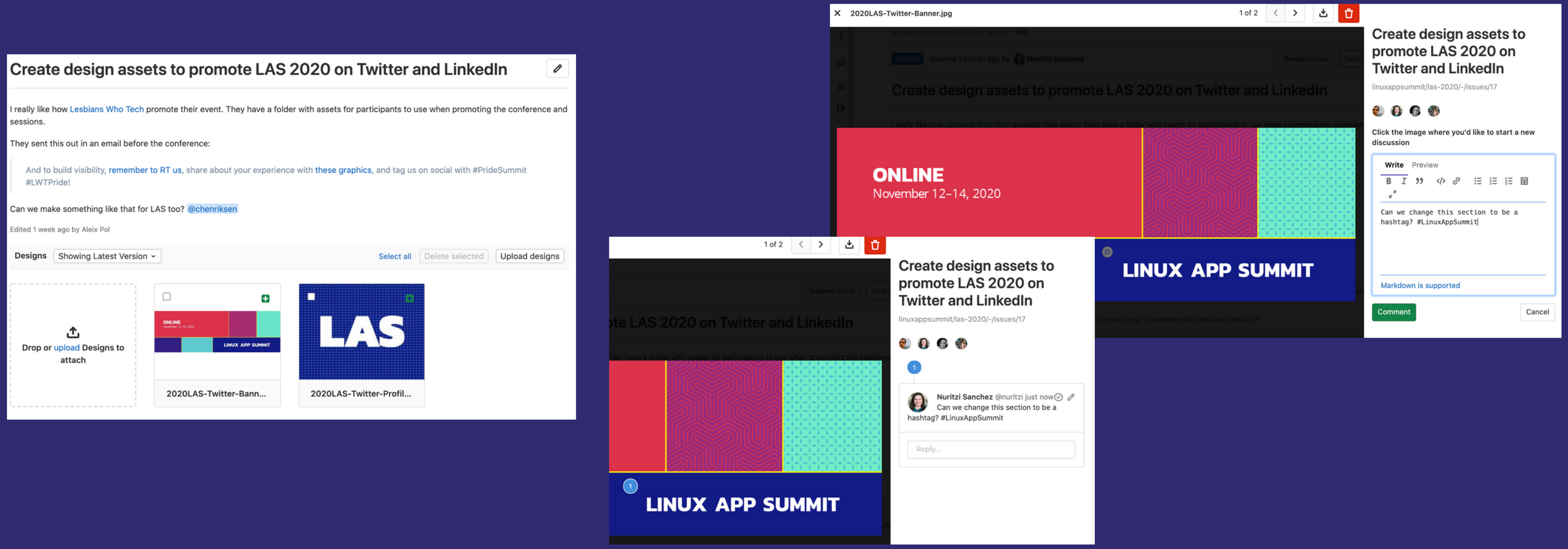
Lists with tasks

`* [ ]` - creates the fancy checkboxes we use to mark tasks done.



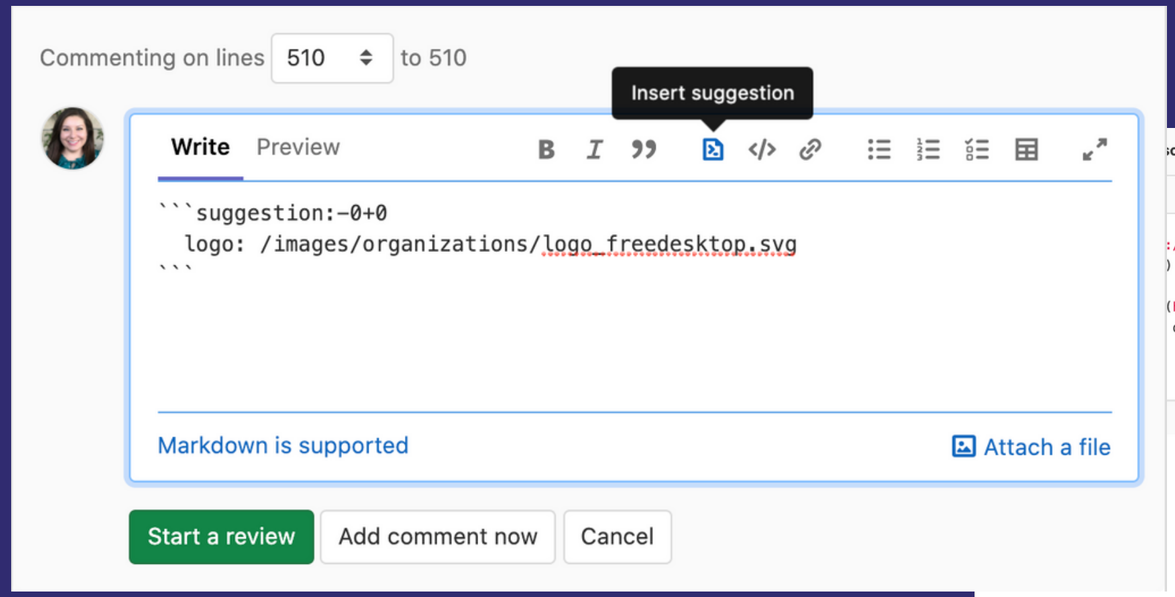
Design management

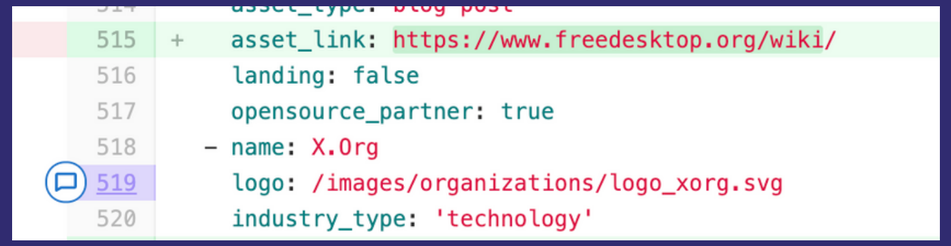
With the recent addition of design management to Core, everyone can contribute even more! These features allow you to upload designs and suggest ideas, thoughts and changes.

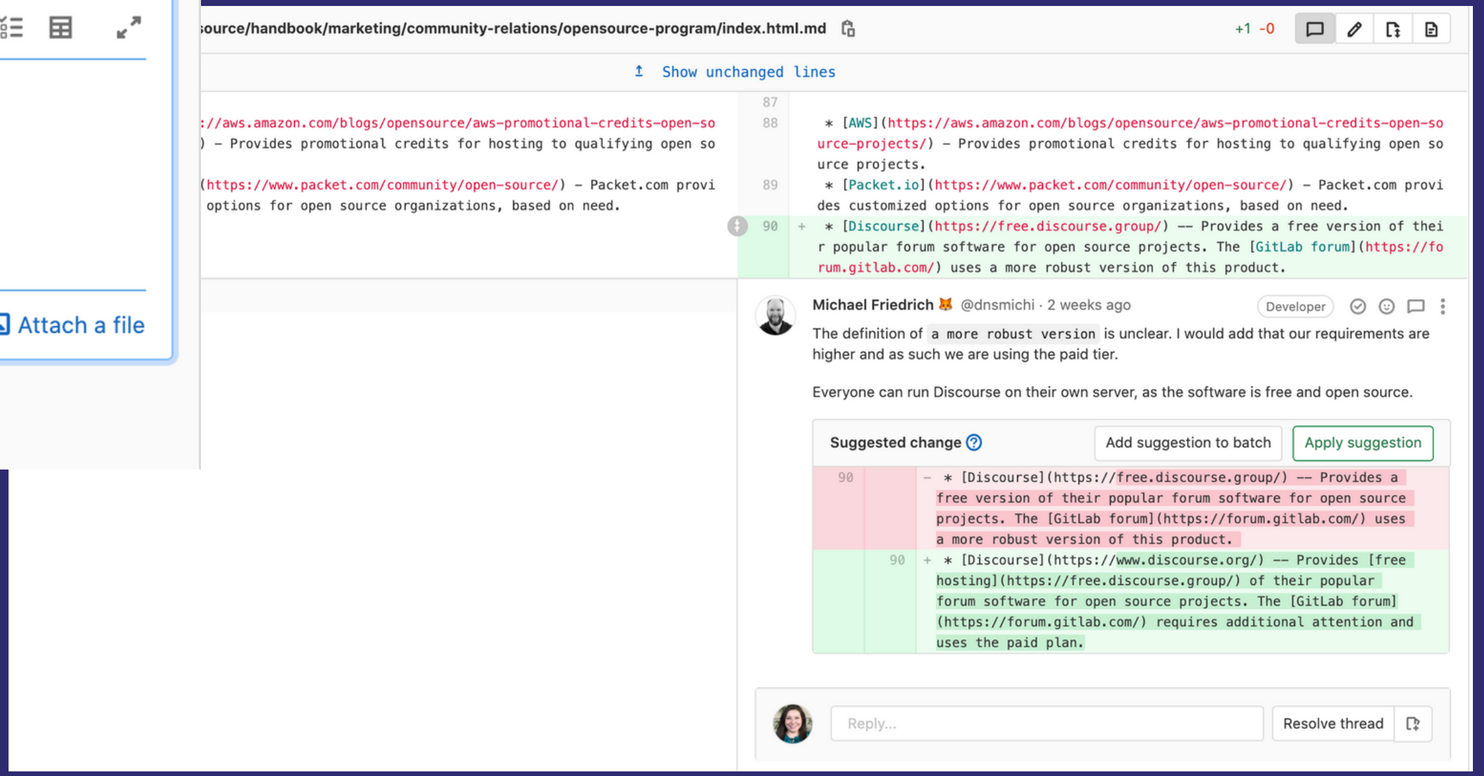


Suggestions in code

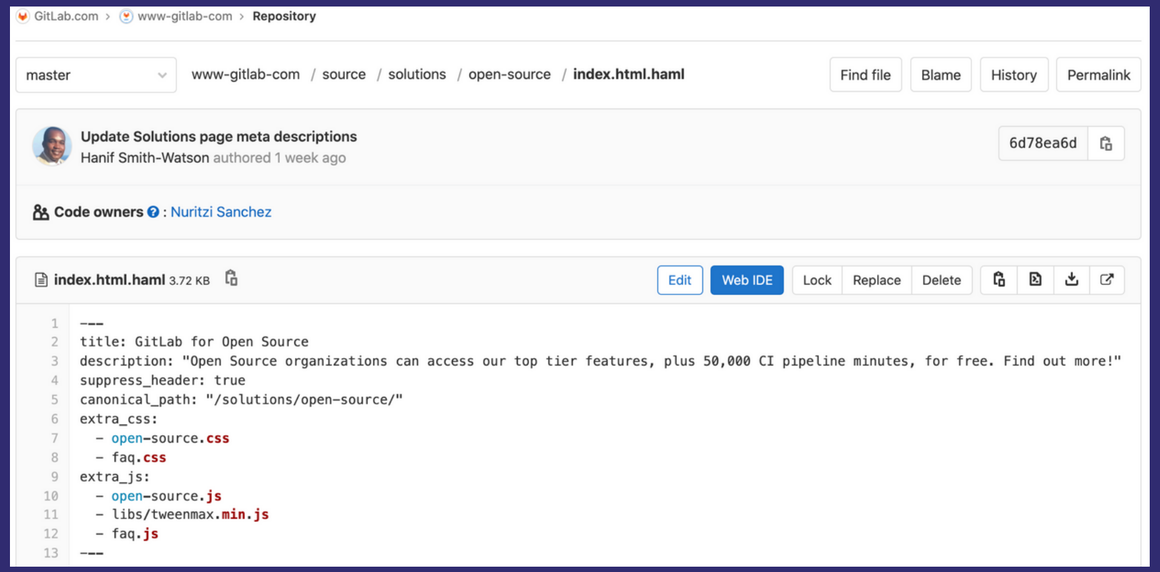
Snippets and commenting in line -- both are super useful!



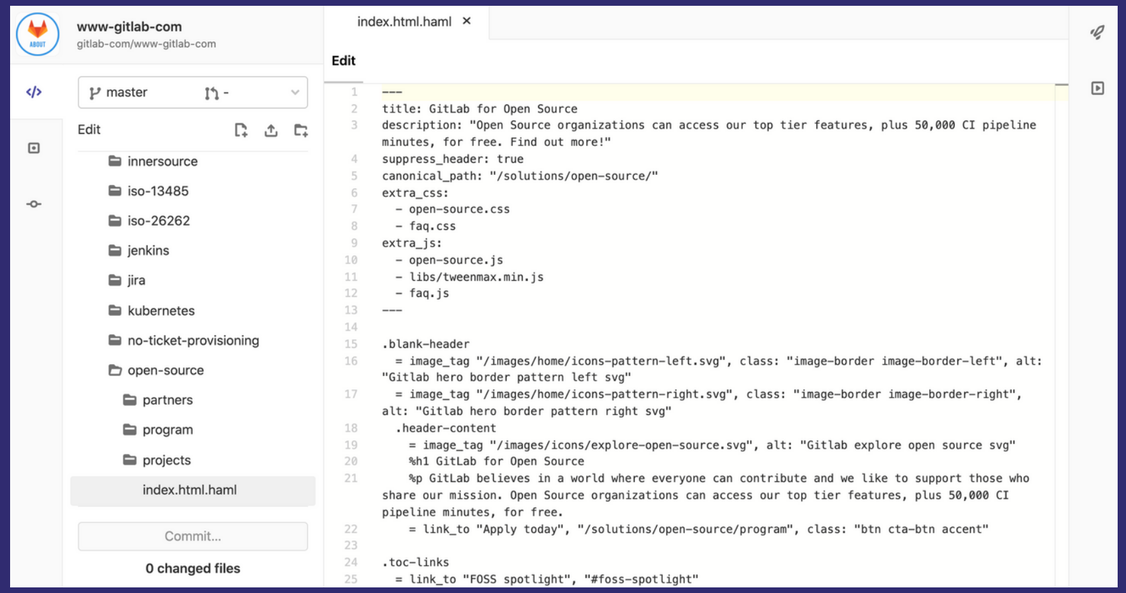




Different ways to make MRs



Edit the markdown page directly



Open the file in the WebIDE

https://about.gitlab.com/blog/2020/05/28/using-gitlab-web-ide-gitlab-ci-cd/

GNOME
GitLab

Thank You!

Nuritzi Sanchez

@1nuritzi

/nuritzis