

Project Kickoff!

CS 346: Application
Development

GitLab setup

What to do before you start coding!

Establish Milestones




Create **milestones** for each of your four software releases.

Plan > Milestones

Jeff Avery > cs346-template > Milestones

Open 3 Closed 0 All 3

Filter by milestone name Due soon ▾ New milestone

Sprint 1 Feb 7, 2022–Feb 18, 2022 Expired Jeff Avery / cs346-template	 6 Issues · 0 Merge requests 100% complete	Close Milestone
Sprint 2 Feb 28, 2022–Mar 11, 2022 Expired Jeff Avery / cs346-template	 3 Issues · 0 Merge requests 66% complete	Close Milestone
Sprint 3 Mar 14, 2022–Mar 25, 2022 Expired Jeff Avery / cs346-template	 2 Issues · 0 Merge requests 50% complete	Close Milestone

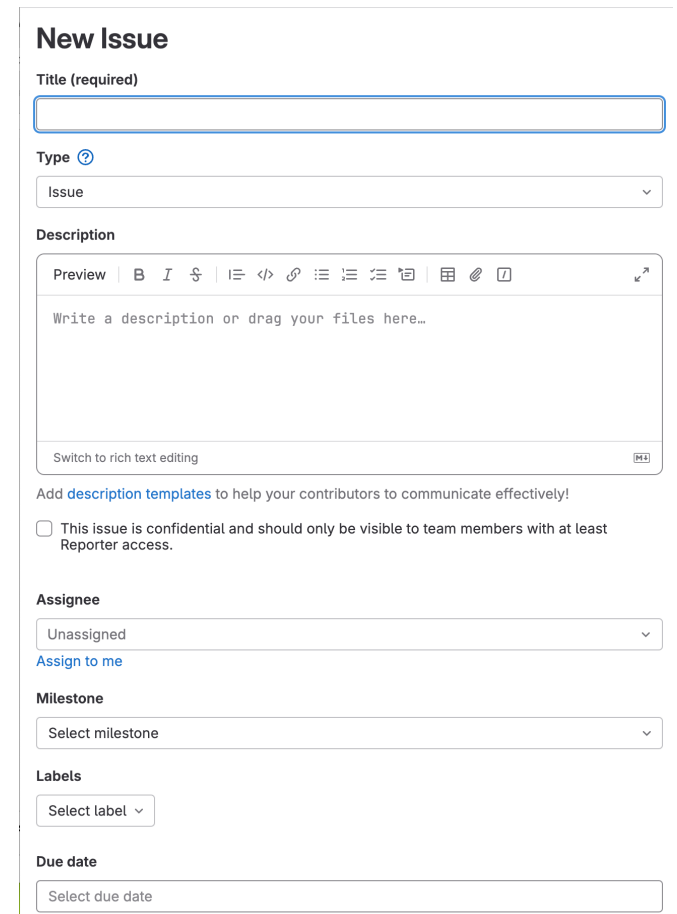
Create Issues

Issues represent the tasks that you need to complete for your project.

- Plan > Issues

Every feature from your solution (design proposal) should be logged as an issue.

- You may need multiple issues per feature!
- Each issue is functionality that can be implemented and tested independently.



The image shows a 'New Issue' form with the following fields and options:

- Title (required):** A text input field.
- Type:** A dropdown menu with 'Issue' selected.
- Description:** A rich text editor with a toolbar (Preview, Bold, Italic, Underline, Link, Code, Unlink, Bulleted List, Numbered List, Indent, Outdent, Image, Table, Link, Unlink) and a text area containing the placeholder 'Write a description or drag your files here...'. A 'Switch to rich text editing' button is at the bottom right.
- Confidentiality:** A checkbox labeled 'This issue is confidential and should only be visible to team members with at least Reporter access.'.
- Assignee:** A dropdown menu with 'Unassigned' selected and a link 'Assign to me' below it.
- Milestone:** A dropdown menu with 'Select milestone' selected.
- Labels:** A dropdown menu with 'Select label' selected.
- Due date:** A dropdown menu with 'Select due date' selected.

Guidelines for creating issues

- Issues should “fall out” from your proposed solution.
 - Clearly link your issues to your user stories and your design proposal.
 - Issues should be created AFTER the design proposal is submitted.
- Each issue should represent about 0.5 days of work (or less).
- Include as much detail as you know. This is the main (only?) source of information on what this work entails.
- All new issues should be
 - Not assigned to a milestone.
 - Not assigned to a person.

Recommended: Labels

Create tags for priority that you can use to label and sort issues.

Manage > Labels

Prioritized labels
Drag to reorder prioritized labels and change their relative priority.

high Jeff Avery / notae	Prioritized	Issues	Merge requests	★	Subscribe	⋮
medium Jeff Avery / notae	Prioritized	Issues	Merge requests	★	Subscribe	⋮
low Jeff Avery / notae	Prioritized	Issues	Merge requests	★	Subscribe	⋮

Kickoff activities

How to get started!

Planning instructions

Make sure everyone agrees on the project direction.

- Review the feedback from the design proposal and address it.
- You can make changes to the plan as the project progresses, but they need to be made as a team (in meetings like this).

Make sure that you have issues logged for each your features (1:n).

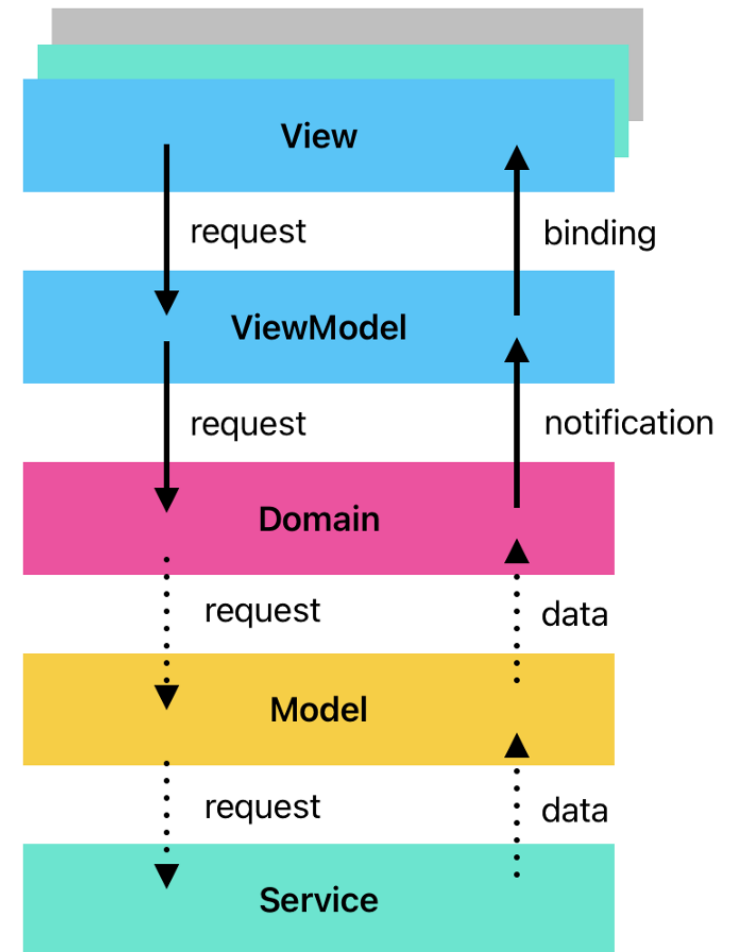
- An issue should describe the problem adequately for someone to investigate/design/implement the solution.

In this meeting, choose which features to implement for this iteration.

<https://student.cs.uwaterloo.ca/~cs346/1251/course-project/deliverables/project-iterations.html>

What to build first?

- Build what you know how to build. You can:
 - Structure your code into layers.
 - Build user interface screens and navigate between them.
 - Build Domain objects e.g., recipe classes, note classes to model your data.
 - Write unit tests for domain and lower.
- As much as possible, build working features. If necessary, “fake” the data.
 - e.g., a Recipe App might launch with a list of recipes and images that are hard-coded (ideally in a mock Model class). You can replace the fake data later when the DB works.



What is the outcome of a kickoff?

At the end of the kickoff:

- All the features (issues) that you intend to complete for the first milestone are assigned to the Demo 1 milestone.
 - Other issues should be logged but should be unassigned.
- Each person should have some of these issues assigned to them.
- You should know when you will next meet.

<https://student.cs.uwaterloo.ca/~cs346/1251/course-project/deliverables/project-iterations.html>

What's next?

Work is assigned. Now what?

Toolchain setup

Check that everyone has the toolchain installed.

- Make sure that everyone in your team is using the same IDE!

Create your project

- Use the project wizard in your IDE, or git-init and modify it.
- Commit and push the project to Git.
- Check that everyone can open/build/run it!

Coordinate your work

Work together as much as you can

- You are required to meet at least twice each sprint in-person.
- Meetings can be in-class, but everyone should attend.
- [Take meeting minutes when you meet](#) – record who attended, date/time, what was discussed/any decisions that were made.

Keep issues up-to-date and write detailed commit messages.

- We will review Git commit logs and GitLab issues to see what each person did.

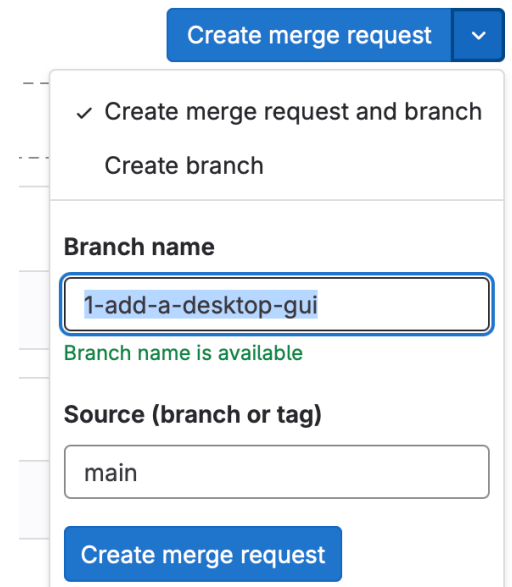
Use feature branches!

You should create a working branch for each feature.

- These type of branches are called **feature branches**.
- Once we have a feature implemented and tested, merge our changes back into the trunk (“integration”).
- GitLab: Create merge request in the issue screen.

A typical workflow would be:

1. Create a feature branch for that feature.
2. Make changes on your branch only. Test everything.
3. (Optional) Have changes code reviewed by a teammate.
4. Switch to `main`; merge your feature branch into `main`.



Create merge request

✓ Create merge request and branch
Create branch

Branch name

1-add-a-desktop-gui

Branch name is available

Source (branch or tag)

main

Create merge request

“Create merge request” will create a feature branch for you.