Setting Up Department Gitlab Repository For Projects Submission

You’ll be submitting your projects on gitlab repository for grading, but first you need to add the TAs as members (a.k.a. contributors) to your repository. You should follow these guidelines to setup a repository, add the TAs and finally, commit your project code.

Setting Up Github

1. Go to gitlab.cs.dartmouth.edu
2. Login (or Register) with your Dartmouth email address
3. After Login, you’ll be redirected to the home page which looks like the snapshot below

   ![Gitlab Home Page Snapshot](image)

   - [Create a project](#): Projects are where you store your code, access issues, wiki and other features of Gitlab.
   - [Create a group](#): Groups are the best way to manage projects and members.
   - [Explore public projects](#): There are 80 public projects on this server. Public projects are an easy way to allow everyone to have read-only access.
   - [Learn more about GitLab](#): Take a look at the documentation to discover all of GitLab’s capabilities.

4. Create a new Project. You should name it as “lastname_firstname_CS65” (e.g Campbell_Andrew_CS65)
5. On the New Project Page, Go to Settings (Last Option on the Navigation Pane)
6. Select Members
7. You can now add the TAs and graders as members (masters) to your project. Once you add us, we’ll be granted access. Here is our Gitlab accounts:
   1. @shayan
   2. @kmasaba

8. Next step is to link your project in Android Studio with this Git project. It’s only then that you can commit your code to this repo.

**Committing Code to Gitlab Repository**

There are two ways by which you can commit your code to Gitlab repository. You can run Git commands in the terminal or you can use the UI of Android Studio. The former is recommended.

Before getting started, create one directory on your computer (e.g., CS65_Assignments) which will be your workspace for all the class projects. So this directory should include “CodeIts”, “MyRuns” and “FinalProject” folders. A sample folder hierarchy is shown below.
Using the Git Commands (Recommended)

1. $ cd /path/to/project_root_folder
2. git init
3. git remote add origin <url to your repo - check attached screenshot>
4. git add .
5. git commit -m "Initial commit"
6. git push -u origin master

Replace <xxx> with the correct values and follow the steps carefully.

After pushing, you can see your first commit on the Gitlab project.

And whenever you have an update that you like to push to the repository, run the following commands.

1. $ cd /path/to/project_root_folder
2. git add .
3. git commit -m "<Some message>"
4. git push

Using Android Studio UI (Not Recommended)

Open the terminal and navigate to the directory you created, and then commit and push your files and folders to sync the existing local directory with the remote repository.

For the “git remote add origin <repo_url>” command, you'll find the <repo_url> on your git project page.
After pushing, you can see your first commit on the Gitlab project.
1. To get started, open Android Studio with your project.
2. Select VCS->Enable Version Control Integration…
3. Choose git as the version control for your Android Studio project

4. Go to Android Studio Preferences, and check whether git is already added to your root folder (e.g. CS65_Assignments)
If it’s not then, add your root folder that you initialized git inside (e.g. CS65_Assignments). Make sure to choose “Git” as “VCS”.

5. To commit your code changes on your local machine or push them to the server, go to VCS -> Commit…
6. Write a message about your updates before committing the changes, and then select commit or push. You can either just Commit or Commit and Push.

7. If you’re going to push code changes to the server, you will be asked to enter your credentials for the remote repository.
Once you push the commit, you will see something like this at the bottom of Android Studio.

8. You can see your last commit on the Gitlab website at your project.
And you are all set! There are tons of material available online about this, and you can always use Piazza if you have questions.