GitLab
Onboarding

Faculty of Computer Science
Fall 2020
What is GitLab

- Git is a revision control system that seems to be dominant, supplanting RCS, CVS, SVN
  - Synchronization of repositories (Local-Server, L-L, BranchFromRepo)
- Faculty of Computer Science has a GitLab Enterprise (Ultimate) Edition\(^{(1)}\) install to manage our local non-public projects
- GitLab provides a number of extra functions beyond a simple git repo
- GitHub is for public projects

\(^{(1)}\) Licensed for Education and Research only
GitLab

• Revision Control
  • Plus Pull Requests, etc.
• Continuous Integration
• Extras:
  • Ticket and Time Tracking
  • Project Wiki
  • Burndown Charts and Productivity Analytics
  • Issues (and inter-project issues) and Issue Boards
  • Iterations and Roadmaps
  • Code Quality Reports and Review Analytics
  • Can connect Jenkins and Jira
  • Cross-Project Dashboards and Analytics
• Visit https://git.cs.dal.ca for gitlab web i/f
  • Default view is all the projects you have access to

• Create repo in gitlab web interface. In an empty project you get directions for using it as a remote in your local dev env

• Upload an ssh key to establish identity
  • (upper right on picture)=> Settings, SSH Keys

• git clone git@git.cs.dal.ca:group/repo.git
  • Note that everything runs as git user, and not csid@gitlab
Class Repositories

• We create repositories for users to build their projects in, where the professor/TA can view the code
  • Courses/Term/Course/Assignment N/CSID.git
    • Admin + Instructor + Student (s/b private)
  • E.g. Courses/2019-Fall/CSCI 2134/Assignment 6/Maxwell/myprojectfiles

• Currently we have a PHP script that builds this structure from the Admin API. There is also a shell+python script for similar purposes.
GROUP PROJECTS WOULD BE BUILT AS COURSE/PROJECT/GROUP (A-E) WITH UNASSIGNED PERMISSIONS AND THEN INSTRUCTOR WOULD GO IN AND ASSIGN STUDENTS TO GROUPS.

RESEARCH GROUPS REPOSITORIES

BIG DATA IS USING A GROUP WITH PROJECT REPOSITORIES
• We have 4 Shared Runners configured
  • **Docker Autoscale in Openstack**
    “dalfcs_docker_autoscale”/”dalfcs_gitlab_docker_ci”
  • **Docker-based single machine**
    “dalfcs_gitlab_docker_ci”
    • Clean environment for each integration, specify Docker image to start with
  • **Timberlea “ugrad”**
    • Run stuff as “runner” user on timberlea with anything installed on Bluenose available
  • **Waverley “grad”**
    • Like ”ugrad” but runs on hector instead

• In quotes is the tag - each job must specify a tag to be caught; no longer have an untagged shared runner.

• Tech Services uses runners to continuously deploy web projects
• Project Limits (per user)
  - Currently, by default, users are limited to 240 projects. If a user requires more, contact the CS Help Desk (cshelp@cs.dal.ca).

• Runners
  - Each of the 3 single-machine CI/CD runners is a serial pipeline, so additional CI's wait for other projects to complete. The auto-scale will keep adding runners until it runs out of Openstack Instances (20).