Submitting Web Code with Hubzero

Erich Huebner

Version 2

October 6, 2014

Table of Contents

Overview

This document is to provide you with the guidelines and processes required for committing web code in the hub environment.

Coding Guidelines

Standards

All code must follow the documentation available here:
https://hubzero.org/documentation/current/webdevs

Code Locations

Code should only to be placed within the defined code repository. Code placed elsewhere on the machine will not be executable.

Data Locations

Small (less than ~100MB) static data may be placed in the appropriate directory within the code repository.
Large (greater than ~100MB) should be placed in an appropriate external directory (/data/…) to prevent wasted hard disk space when the code repository creates versions.

Script Locations

Scripts should be placed in an appropriate directory within the code repository.

Portable code

Always use a relative paths, environment variables, and DNS entries to reference files to ensure your project does not break when your code is migrated to another machine during maintenance or upgrades.

Use the HUBzero infrastructure

Coding inside an infrastructure can be very different than coding for a standalone project. Please become familiar with what the infrastructure provides you and what you can do with it. For example, jquery is included. Please use the provided version to avoid conflicts.

Functions and variables

Many functions and variables are provided to ensure security and passing of data within the infrastructure.

No header, footers

HTML header, footer, and body tags are already part of the HUB template and are not necessary.

Database updates

Use “muse” to create and deploy database migration scripts to update the database required by your code. See https://hubzero.org/documentation/1.3.0/webdevs/migrations .

Git

Git is a free and open source distributed version control system designed to handle everything from small to very large projects with speed and efficiency. http://git-scm.com/

Git is used in the HUB environment and it is recommended to become familiar with it as you will be committing code via git. http://git-scm.com/documentation

Code for the production environment

Available packages may not be the latest version available on the web. Please check the version of the packages you require are available. HUBzero web hosts are generally Debian Wheezy (September 2014).

Request for Packages and libraries

Additional packages or newer versions of packages can be requested and we will try our best to install them. However, some packages may not compile. Please allow extra time to custom libraries to be investigated and installed.

Be mindful of security

It is recommend following along with Debian security updates that get applied to the hub OS environment by subscribing to the mailing-list here: https://lists.debian.org/debian-security-announce/

The stage and dev sites are updated with security patches on the night after the security issue is reported as patched (via debian security repo) and the production site follows shortly after pending no major adverse effects to the dev system. Account credentials should never be in code and should be placed in a configuration file on the host and outside the code repository.

Access

Production - Restricted

No access to the production web server will be allowed. The only exceptions are system administrators and lead web developers are allowed in order to apply emergency fixes, which should be very very rare.

Stage - Restricted

No access to the stage web server. The only exceptions are system administrators, and HUBzero web developers are allowed in order to apply template fixes and other adjustments necessary for custom hub items (view overrides).

Dev – Semi-Managed

Non-HUBzero personnel are allowed access to the HUB web server and it's database. HUBzero system administrators will administer security updates as necessary. HUBzero developers will not have access. Web code management is the sole responsibility of the non-HUBzero personnel granted access.

Personal VM – Self-managed

Non-HUBzero personnel's individual development environment based on the HUBzero VM. HUBzero system administrators and HUBzero developers will not have access.

Code Flow - HUB

VM / Dev

Code should be developed here - HUB components, modules, and plugins *not for super groups). Please make sure you have the most update-to-date code from stage as you will not be able "git push" from a terminal there.

Check-in code.

1)"git status" to see what files have been added or modified. **There is one repository for web code, please check with other developers before committing their code and provide and appropriate commit message if working in a common dev envrionment.

2)"git commit" from the git repo folder will prompt your for a commit message. The commit message must be prefixed with [HUBNAME][COMPONENT NAME].

Push code to stage for review.

1)"git push" from a terminal.

Stage

Code will be scanned and reviewed here for standards and security. This host may be used for testing and demonstrations before promoting to production. Code can only be promoted to production when all stakeholders of projects agree it is time to publish new versions of projects that wish to be updated (**There is one repository for web code). Testing must also be completed. Submit a request to promote code to production at least one week (5 business days) prior to the due date.

Production

Code here is thoroughly testing and stable. For updates here, please start in dev.

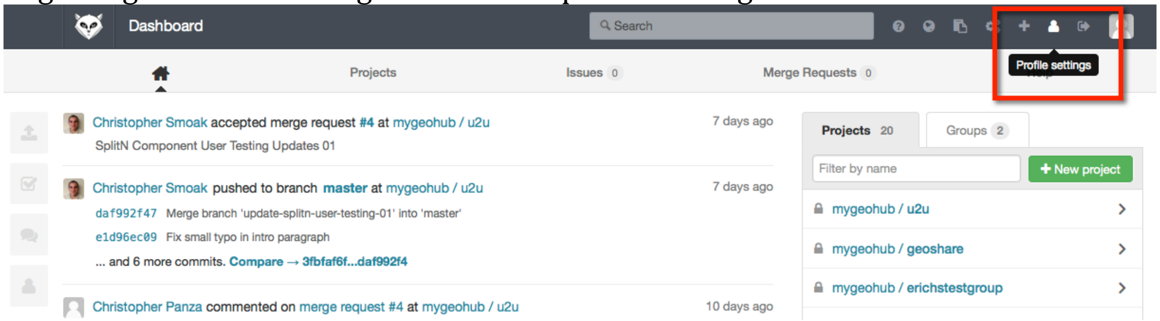
Code Flow – Super Groups

Using Gitlab (https://gitlab.hubzero.org)

Gitlab notifications

You may want to set your Gitlab notifications to "Watch" so you will receive all notifications from projects in which you participate.

Log into gitlab.huzbero.org and click on "profile settings".

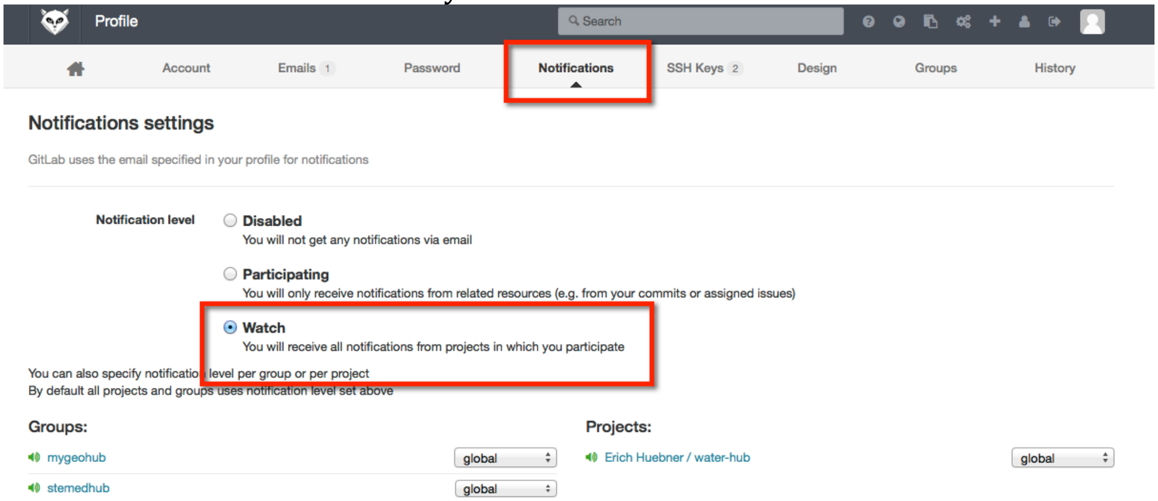


Christopher Smoak accepted merge request #4 at mygeohub / u2u 7 days ago
SplitN Component User Testing Updates 01

Christopher Smoak pushed to branch master at mygeohub / u2u 7 days ago
daf992f47 Merge branch 'update-splitn-user-testing-01' into 'master'
e1d96ec09 Fix small typo in intro paragraph
... and 6 more commits. Compare → 3fbfaf6f...daf992f4

Christopher Panza commented on merge request #4 at mygeohub / u2u 10 days ago

Click on the notification tab and you should see a screen like this:



Notifications settings

GitLab uses the email specified in your profile for notifications

Notification level

○ **Disabled**
You will not get any notifications via email

○ **Participating**
You will only receive notifications from related resources (e.g. from your commits or assigned issues)

● **Watch**
You will receive all notifications from projects in which you participate

You can also specify notification level per group or per project
By default all projects and groups uses notification level set above

Groups:
mygeohub global
stemedhub global

Projects:
Erich Huebner / water-hub global

Then click on the radio selector to the left of "Watch" for the notification level. There isn't a save button as your selection saves automatically. You may now log out or go to another page within Gitlab.

VM / Dev

Code should be developed here – Super Group components, modules, and plugins (*not for overall HUB). Make sure you have the most update-to-date code from GitLab (*required) to prevent manual merges in GitLab.

Fork and Clone.

See https://hubzero.org/documentation/current/webdevs/supergroups_gitlab

For reference: https://www.atlassian.com/git/tutorials/comparing-workflows/forking-workflow . Thanks CGP.

Push code to Gitlab.

https://hubzero.org/documentation/current/webdevs/supergroups_gitlab

Request Merge.

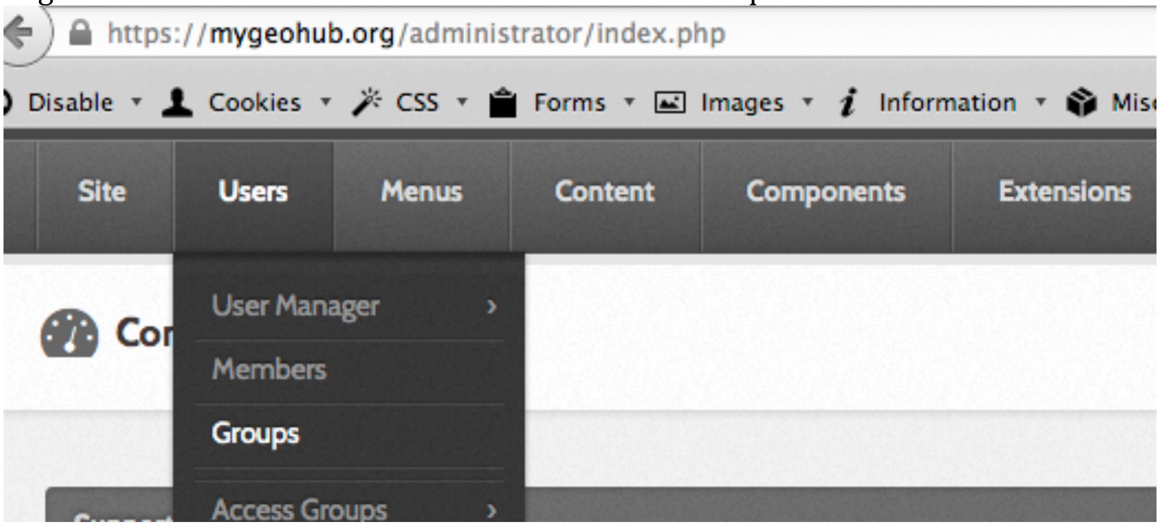
Please, submit a merge request (Gitlab)to promote code to stage at least one week (5 business days) prior to the due date.
Details: https://hubzero.org/documentation/current/webdevs/supergroups_gitlab

Coordinate with your team lead.

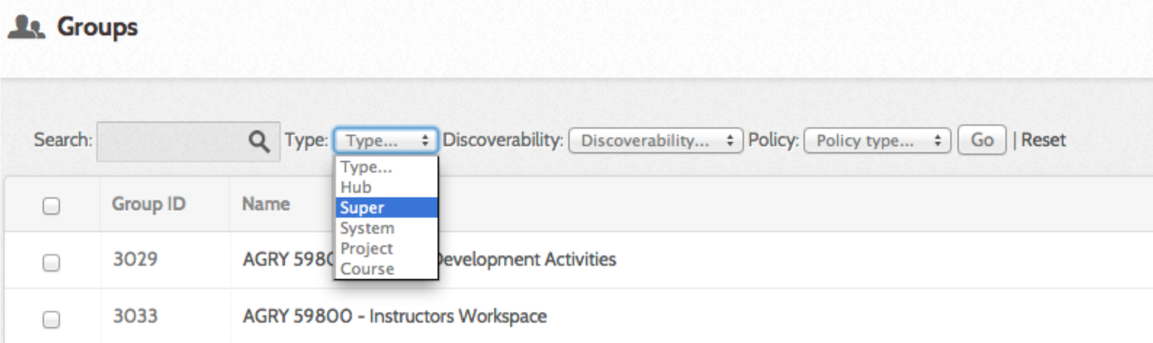
After a merge request has been reviewed and granted by HUBzero, your team leader (or project leader) is required to pull the new code into the appropriate group for coordination purposes.
https://hubzero.org/documentation/1.3.0/webdevs/supergroups_gitlab.developing

To pull in code to stage or production (for team leaders):

Log into the administrator interface and click on "Groups" from the "Users" menu.

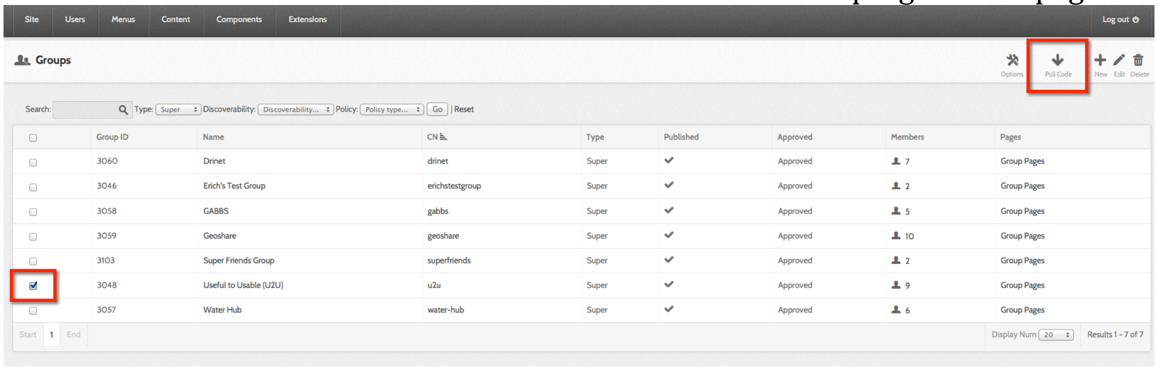


Find the super group that you wish to update. There are a few way to do this, but I found the easiest is to sort by type "super". Choose the type from the drop down selector and then click "Go".



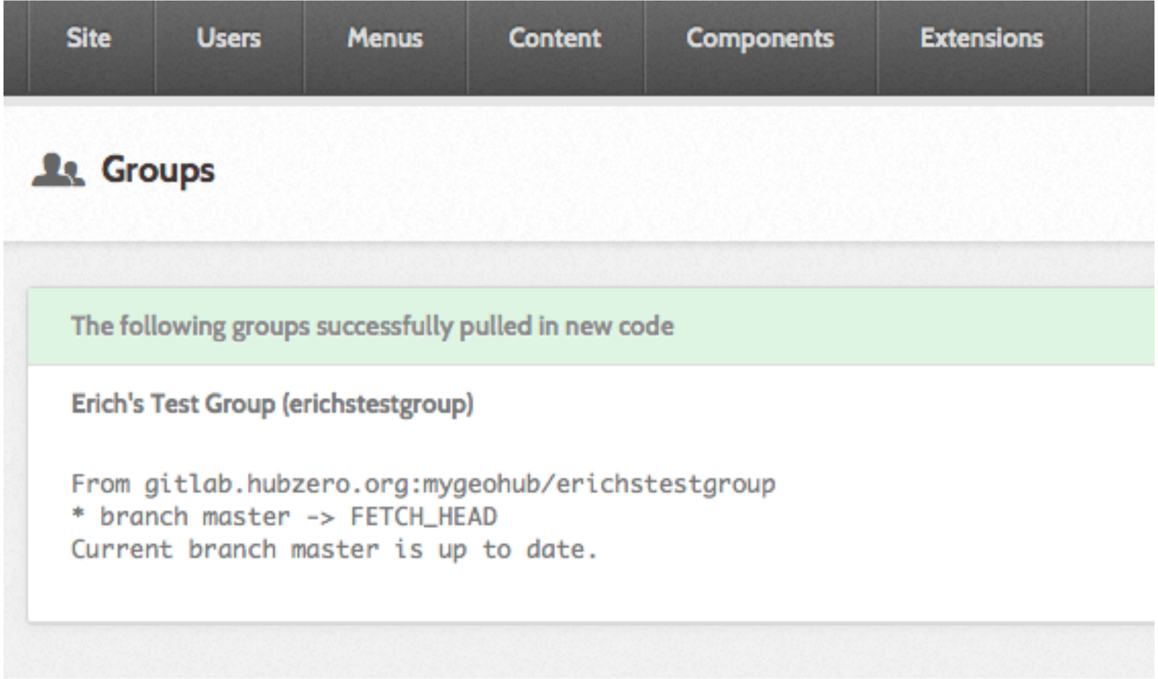
☐	Group ID	Name
☐	3029	AGRY 598 Development Activities
☐	3033	AGRY 59800 - Instructors Workspace

Check the box next to the super group that you wish to pull in new code from Gitlab and then click on "Pull code" from the action buttons at the top right of the page.



☐	Group ID	Name	CN	Type	Published	Approved	Members	Pages
☐	3060	Drinet	drinet	Super	✓	Approved	7	Group Pages
☐	3046	Erich's Test Group	erichstestgroup	Super	✓	Approved	2	Group Pages
☐	3058	GABBS	gabbs	Super	✓	Approved	5	Group Pages
☐	3059	Geoshare	geoshare	Super	✓	Approved	10	Group Pages
☐	3103	Super Friends Group	superfriends	Super	✓	Approved	2	Group Pages
☑	3048	Useful to Usable (U2U)	u2u	Super	✓	Approved	9	Group Pages
☐	3057	Water Hub	water-hub	Super	✓	Approved	6	Group Pages

You should now see a page similar to this:



The following groups successfully pulled in new code

Erich's Test Group (erichstestgroup)

```
From gitlab.hubzero.org:mygeohub/erichstestgroup
* branch master -> FETCH_HEAD
Current branch master is up to date.
```

The process is complete and the code is updated.

Stage

Code will be scanned and reviewed here for standards and security. This host may be used for testing and demonstrations before promoting to production. Code can only be promoted to production when all stakeholders of projects agree it is time to publish new versions of projects that wish to be updated (**There is one repository for web code). Testing must also be completed. Please, submit a request (hub support ticket) to promote code to production at least one week (5 business days) prior to the due date.

Production

Code here is thoroughly testing and stable. For updates here, please start in dev.

Updating dev/VM with new hubzero code

Change directory to the web root repo, usually /www/dev and run:
“git pull --rebase”
**This must be setup by HUBzero for personal VMs.