Continuous Integration and Code Review: how IT can help

Alex Lossent – IT/PES – Version Control Systems

Version Control Systems

- Primary role:
  - provide means to manage changes to source code
- Other services to developers:
  - Documentation: Twiki
  - Issue Tracking: Jira
  - Code Review: GitLab
  - Continuous Integration: Jenkins

A Code Review tool: why?

- Ensure the quality of new features/fixes
- Provide means to discuss and comment on code changes
  - E.g. decide if the code is ready for production
- Better manage concurrent work and merge conflict resolution
- Simplify permissions
- Enable contributions from outside the core dev team
  - without risk for the main code repository

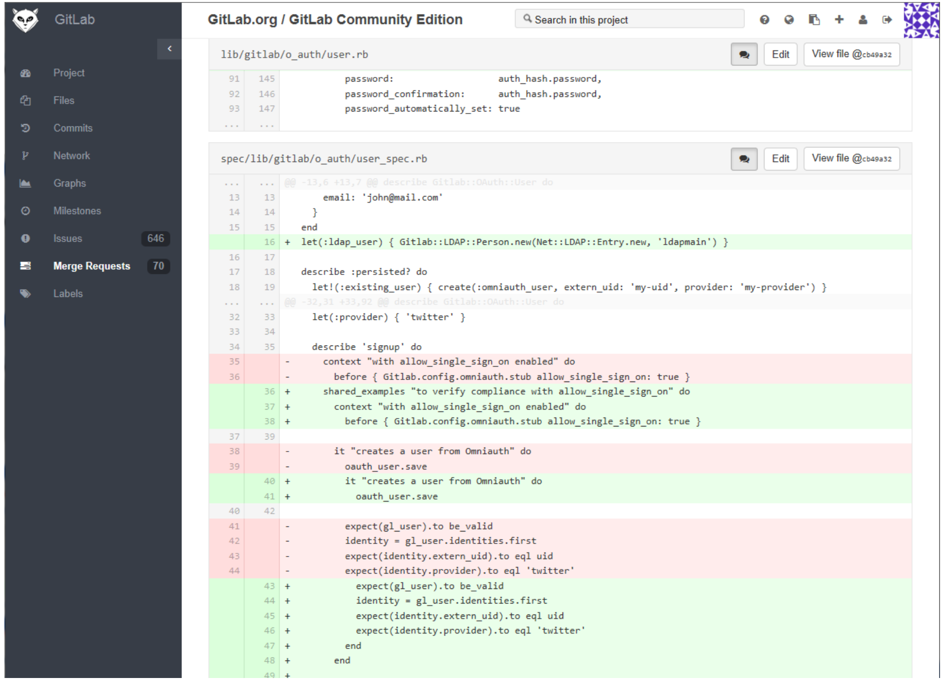
GitLab

- New hosting platform for Git repositories
  - On-premise “GitHub lookalike”
- CERN users only (like SVN/git.cern.ch)
  - GitHub remains appropriate for open-source/external collaborations (KB0003132)
- Recommended for any new software project

Code review workflow

- Important branches are protected
- Changes (fix, new feature…) written in feature branches or project forks
  - Then submitted as Merge Requests to the main branch
- Merge Requests can be reviewed before being merged
  - Discuss, comment on code, propose new changes, vote, etc.
  - “master” developers eventually click the “merge” button

Merge requests

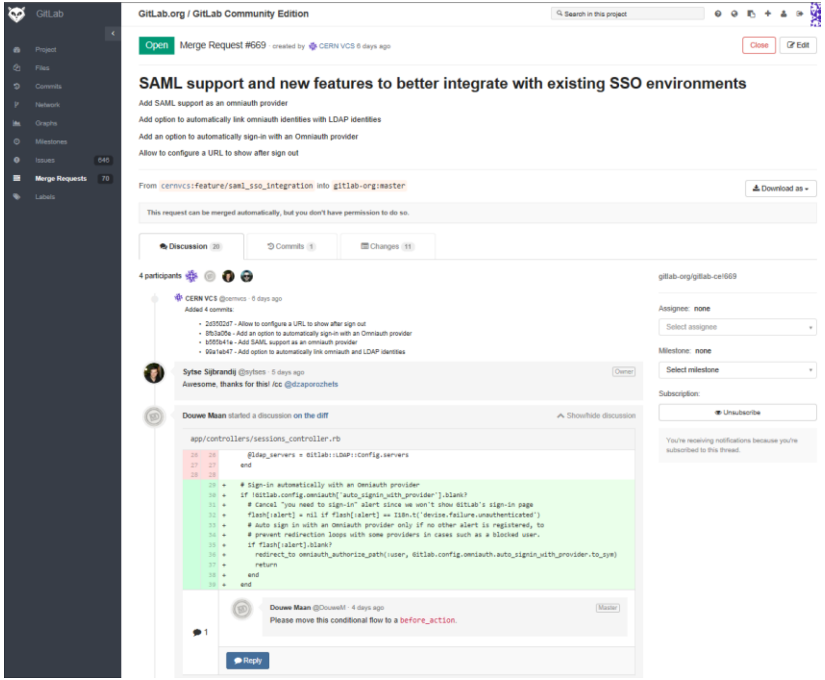


lib/gitlab/o_auth/user.rb

```
          password:                   auth_hash.password,
          password_confirmation:      auth_hash.password,
          password_automatically_set: true
```

spec/lib/gitlab/o_auth/user_spec.rb

```
 @@ -13,6 +13,7 @@ describe Gitlab::OAuth::User do
       email: 'john@mail.com'
     }
   end
+  let(:ldap_user) { Gitlab::LDAP::Person.new(Net::LDAP::Entry.new, 'ldapmain') }
 
   describe :persisted? do
     let!(:existing_user) { create(:omniauth_user, extern_uid: 'my-uid', provider: 'my-provider') }
 @@ -32,31 +33,92 @@ describe Gitlab::OAuth::User do
     let(:provider) { 'twitter' }
 
     describe 'signup' do
-      context "with allow_single_sign_on enabled" do
-        before { Gitlab.config.omniauth.stub allow_single_sign_on: true }
+      shared_examples "to verify compliance with allow_single_sign_on" do
+        context "with allow_single_sign_on enabled" do
+          before { Gitlab.config.omniauth.stub allow_single_sign_on: true }
 
-        it "creates a user from Omniauth" do
-          oauth_user.save
+          it "creates a user from Omniauth" do
+            oauth_user.save
 
-          expect(gl_user).to be_valid
-          identity = gl_user.identities.first
-          expect(identity.extern_uid).to eql uid
-          expect(identity.provider).to eql 'twitter'
+            expect(gl_user).to be_valid
+            identity = gl_user.identities.first
+            expect(identity.extern_uid).to eql uid
+            expect(identity.provider).to eql 'twitter'
+          end
+        end
+
```



Merge Request #669 · created by CERN VCS 6 days ago

SAML support and new features to better integrate with existing SSO environments

Add SAML support as an omniauth provider

Add option to automatically link omniauth identities with LDAP identities

Add an option to automatically sign-in with an Omniauth provider

Allow to configure a URL to show after sign out

From `cernvcs:feature/saml_sso_integration` into `gitlab-org:master`

This request can be merged automatically, but you don't have permission to do so.

4 participants

CERN VCS @cernvcs · 6 days ago

Added 4 commits:

- 2d3502d7 - Allow to configure a URL to show after sign out
- 8fb3a06e - Add an option to automatically sign-in with an Omniauth provider
- b585b41e - Add SAML support as an omniauth provider
- 99a1eb47 - Add option to automatically link omniauth and LDAP identities

Sytse Sijbrandij @sytses · 5 days ago

Awesome, thanks for this! /cc @dzaporozhets

Douwe Maan started a discussion on the diff

app/controllers/sessions_controller.rb

```
       @ldap_servers = Gitlab::LDAP::Config.servers
     end
 
+    # Sign-in automatically with an Omniauth provider
+    if !Gitlab.config.omniauth['auto_signin_with_provider'].blank?
+      # Cancel "you need to sign-in" alert since we won't show GitLab's sign-in page
+      flash[:alert] = nil if flash[:alert] == I18n.t('devise.failure.unauthenticated')
+      # Auto sign in with an Omniauth provider only if no other alert is registered, to
+      # prevent redirection loops with some providers in cases such as a blocked user.
+      if flash[:alert].blank?
+        redirect_to omniauth_authorize_path(:user, Gitlab.config.omniauth.auto_signin_with_provider.to_sym)
+        return
+      end
+    end
```

Douwe Maan @DouweM · 4 days ago

Please move this conditional flow to a `before_action`.

1

gitlab-org/gitlab-ce!669

Assignee: none

Milestone: none

Subscription:

You're receiving notifications because you're subscribed to this thread.

Benefits w.r.t. code quality

- Better control on what gets merged into the main branch (and when)
- Enables peer review
  - Find bugs early
  - Enforce coding standards, consistent design
  - Knowledge sharing
- Feature branches allow consistent testing before merge

Other features

- Hierarchical project (= repository) organization
  - Groups – contain multiple projects for a given team
  - Personal workspace – removed when user leaves CERN, a la home directory
- Code browsing
  - Including editing files from the web interface
- Self-service
  - One-click project creation, repository immediately available to work with
- Streamline integration with related services:
  - Issue Tracking, Continuous Integration, Documentation…
- SSH support & SSH key authentication

Getting started with GitLab

- https://gitlab.cern.ch
- KB0003137 “Getting started with GitLab”
- Git Service for support
- Migrate from SVN or git.cern.ch to GitLab

Continuous Integration

- Goal: build better quality software, faster
- How?
  - Developers integrate their work frequently
    - I.e. push code
  - An automated process applies quality control
    - May cover compilation, testing, staging, deployment, doc generation…

What can IT offer in terms of CI?

- Our aim: enable software projects to easily deploy Continuous Integration
  - Documentation
  - Means to provision a CI platform with minimal effort
  - Integration with GitLab
- Out of scope:
  - Providing a CI service
  - Integrate existing CI setups

Current situation

- Since end 2014, we provide a CI "platform-as-a-service"
  - Using well-established, open-source Jenkins
- Clear interest from developers
  - ~40 instances as of September 2015
- A "platform" rather than a "service"
  - We provide the infrastructure (servers), pre-configured SSO and e-group integration
  - But configuring & managing Jenkins itself is entirely left to users (no support)
- Full details presented at ITTF 5-Dec-2014

Jenkins & GitLab

- Jenkins can work together with GitLab
  - Trigger build/test jobs whenever code is pushed
  - Show build/test status on Merge Request page
- https://gitlab.cern.ch/help/integration/jenkins.md

Challenges

- Jenkins is complicated, has lots of plug-ins
- Current model doesn't scale well
  - With # of projects using CI (lots of VMs, manual steps)
- Flexibility could be improved
  - Need to provide your own "slave" nodes if specific libraries/RPMs/OSes are needed by your jobs
  - Some Jenkins plugins are difficult to set up in this model
- We'd like to provide more "added value"
  - Maximize flexibility and customization options
  - Minimize cost of entry and maintenance effort

Outlook for Continuous Integration

- Jenkins Platform-as-a-Service using Docker/Kubernetes/Openshift
  - A genuine PaaS system, fully self-service
  - Better resource efficiency, scalability
  - Resolve mishaps in current implementation (SSO…)
  - Docker images customizable at will
- gitlab-ci (just merged into GitLab 8.0)
  - Lightweight, excellent integration with GitLab
  - CI settings under version control (a la travis-ci)

Getting started with CI

- Getting your own Jenkins instance
  - Request from http://cern.ch/forge
  - Documentation http://cern.ch/jenkinsdoc
- Use the Developers@CERN community
  - Share experience
  - Ask questions to other CI users…
  - https://social.cern.ch/community/developers/

Feedback and input welcome

- How can we make these services more useful?
- Informally via the Developers@CERN community
- Or via the Service Portal
  - Git Service for GitLab
  - Continuous Integration with Jenkins

CERN

www.cern.ch