GitLab vs Bitbucket

**Decision Kit**

**Download Comparison**

<table>
<thead>
<tr>
<th>Feature</th>
<th>GitLab</th>
<th>Bitbucket</th>
<th>Missing in Bitbucket</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manage</td>
<td>5.5/8</td>
<td>1/8</td>
<td></td>
</tr>
<tr>
<td>Plan</td>
<td>6/8</td>
<td>1/8</td>
<td></td>
</tr>
<tr>
<td>Create</td>
<td>7.5/8</td>
<td>5/8</td>
<td></td>
</tr>
<tr>
<td>Verify</td>
<td>6/8</td>
<td>1/8</td>
<td></td>
</tr>
<tr>
<td>Package</td>
<td>4.5/6</td>
<td>1/7</td>
<td></td>
</tr>
<tr>
<td>Secure</td>
<td>7/8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Release</td>
<td>7/8</td>
<td>1/8</td>
<td></td>
</tr>
<tr>
<td>Configure</td>
<td>4.5/7</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monitor</td>
<td>5/8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Defend</td>
<td>1.5/3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Subversion
- Audit Events
- Container Registry
- Compliance Management
- Issue Tracking
- Kanban Boards
- Time Tracking
- Wiki
- Source Code Management
- Code Review
- Static Site Checker
- Continuous Integration
- Code Quality
- Unit Test Coverage
- Load Testing
- Package Repository
- Container Registry
- Hash Check Registry
- Dependency Proxy
- SAST
- Debt
- Code Testing
- Dependency Scanning
- Continuous Delivery
- Pages
- Release Agents
- Advanced Environments
- Auto Env
- Submodule Configuration
- Change Log
- Kubernetes
- Monitors
- Alert Management
- Access Management
- Logging
- Web Application Firewall
- Container Host Security
- Container Network Security
- Code Analytics
- DevOps Reports
- DevOps Management Insights
- Roadmaps
- Service Desk
- Requirements Management
- Quality Management
- Web UI
- Live Preview
- Snapshots
- Design Management
- Web Performance
- Container Networking
- Merge Trains
- Joyner Notebooks
- GSLC
- Dependency Firewall
- Container Scanning
- License Compliance
- Sensor Detection
- Vulnerability Management
- Feature Flags
- Release Orchestration
- Release Evidence
- Security Management
- Serverless Infrastructure as Code
- Cluster Rollout Optimization
- Monitoring
- User Tracking
- DevOps Analytics
- Synthetic Monitoring

**The Decision Kit**

Additional comparisons and detailed information on both products

**View Decision Kit**

**The GitLab Advantage**

Complete DevSecOps platform delivered as a single application.
**Single Application**
One interface, one user model, one data model, easy to use.

**Built-in Security**
Shift left: security testing at the point of code commit.

**Rapid Innovation**
Uniquely transparent product development process. Customers, partners and community can also contribute.

---

**GitLab in Action**
Short live demos that get right to the point.

**GitLab CI for Visibility and Collaboration**

**Foster Collaboration**

**Benefits of a Single App**

---

**Join us on a webinar**

**T-Mobile Customer Spotlight**

Register now

Learn How T-Mobile is Transforming its Business with GitLab on its DevOps Journey.

---

**5 star peer reviews on G2**
Reviews by users across a broad spectrum of companies.

---

**Case Studies**

axway

EAB

esa
## Feature Comparison

<table>
<thead>
<tr>
<th>FEATURES</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Free CI/CD with shared or personal Runners</td>
<td><img src="image" alt="GitLab" /></td>
<td><img src="image" alt="Bitbucket" /></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>GitLab.com has shared Runners that allow you to use GitLab CI/CD completely free up to 400 build minutes for private projects and 50000 minutes for public projects. Alternatively, you can set up your own Runner for faster build processing, unlimited build minutes, or special requirements.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Explore GitLab.com offerings</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Built-in CI/CD</td>
<td><img src="image" alt="GitLab" /></td>
<td><img src="image" alt="Bitbucket" /></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>GitLab has built-in Continuous Integration/Continuous Delivery for free, no need to install it separately. Use it to build, test, and deploy your website (GitLab Pages) or weapp. The job results are displayed on merge requests for easy access.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learn more about CI/CD</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Innerrsourcing</td>
<td><img src="image" alt="GitLab" /></td>
<td><img src="image" alt="Bitbucket" /></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Internal projects in GitLab allow you to promote innerrsourcing of your internal repositories.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Find out more about innerrsourcing</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Commit graph and reporting tools</td>
<td><img src="image" alt="GitLab" /></td>
<td><img src="image" alt="Bitbucket" /></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>GitLab provides commit graphs and reporting tools about collaborators’ work.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learn more about commit graphs</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Availability</td>
<td><img src="image" alt="GitLab" /></td>
<td><img src="image" alt="Bitbucket" /></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>GitLab.com is at 99.9% availability while we want to be above 99.99%. We're hiring to improve this in the last two months of 2017 and in 2018.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GitLab.com status</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The most comprehensive import feature set</td>
<td><img src="image" alt="GitLab" /></td>
<td><img src="image" alt="Bitbucket" /></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>GitLab can import projects and issues from more sources (GitHub, Bitbucket, Google Code, FogBugz, Gitea and from any Git URL) than GitHub or any other VCS. We even have you covered for your move from SVN to Git with comprehensive guides and documentation.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Making it easier to get up and running with GitLab</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Issues</td>
<td><img src="image" alt="GitLab" /></td>
<td><img src="image" alt="Bitbucket" /></td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Quickly set the status, assignee or milestone for multiple issues at the same time or easily filter them on any properties. See milestones and issues across projects.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Learn more about the Issue Tracker</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

*If you want to speed up the delivery cycle, you need to simplify your ecosystem.*

"*GitLab is vastly outpacing SCM competitors such as Bitbucket.*"

"*Our team develops software, and we are also flying spacecraft.*"
### Milestones

Create and manage milestones at both the project and group levels, viewing all the issues for the milestone you're currently working on, representing an Agile program increment or a release.

Learn more about Milestones

### Iterations

Create and manage iterations at the group level, view all the issues for the iteration you're currently working on within your group or project, and enable all subgroups and projects to stay in sync on the same cadence.

Learn more about Iterations

### Confidential Issues

Keep your information secure with Confidential issues. With GitLab, you can create confidential issues visible only for project members with Reporter access level or above.

Learn more about Confidential Issues

### Issue Dependencies

Explicitly mark issues as blocked and blocking and track their status. Blocked issues are visible in the issue card view for easy identification.

Learn more about Issue Dependencies

### Related Issues

Mark issues as related to one another.

Learn more about Related Issues

### Move Issue to Another Project

You can move issues between projects in GitLab. All links, history, and comments will be copied and the original issue will reference the newly moved issue. This makes working with multiple issue trackers much easier.

Learn more about moving issues between projects

### Mark Issue as Duplicate

Mark an issue as a duplicate of another issue, closing it.

Learn more about marking duplicate issues

### Quick Actions

GitLab provides a convenient way to change metadata of an issue or merge request without leaving the comment field with quick actions.

Documentation about quick actions

### Rich Object Summary on Link Hover

View an information-rich summary by hovering over links to users, issues, merge requests, and other objects in GitLab.

See the epic that implements this

Create GitLab Branch from Jira Development Panel
Create a GitLab branch from within the development panel of a JIRA issue.

See issue

Create GitLab Merge Request from Jira Development Panel

Create a GitLab merge request from within the development panel of a JIRA issue.

See issue

Project Issue Board

GitLab has Issue Boards, each list of an Issue Board is based on a label that exists in your issue tracker. The Issue Board will therefore match the state of your issue tracker in a user-friendly way.

Learn more about GitLab Issue Boards

Time Tracking

Time Tracking in GitLab lets your team add estimates and record time spent on issues and merge requests.

Learn more about Time Tracking

Required Merge Request Approvals

When a project needs multiple sign-offs, you can require every merge request to be approved before merging. With Required Merge Request Approvals you can set the number of necessary approvals and predefine a list of specific approvers. In turn, guarantee the quality and the standards of your code.

Learn more about merge request approvals

Multiple approvers in code review

In GitLab, to ensure strict code review, you can require a minimum number of users to approve of a merge request before it is able to be merged. You can undo an approval by removing it after the fact.

Approvals Documentation

Approval rules for code review

Make sure the right people review merge requests with approval rules by specifying lists of eligible approvers, the minimum number of approvals for each, and which target branches they protect. This makes it easy to request review from different teams like Engineering, UX and Product.

Approvals Documentation

Repository pull mirroring

Mirror a repository from a remote Git server to your local server, making it easy to keep local forks and replicas up to date.

Learn more about repository pull mirroring

Optional Merge Request Approvals

Code review is an essential practice of every successful project, and giving your approval when a merge request is in good shape is an important part of the review process, as it clearly communicates the ability to merge the change.

Learn more about optional merge request approvals

Wiki based project documentation
A separate system for documentation called Wiki, is built right into each GitLab project. Every Wiki is a separate Git repository.

Learn more about GitLab Wikis

Design Management

Design Management allows users to upload design assets (such as wireframes and mockups) to GitLab issues and keep them stored in one single place, giving product designers, managers, and engineers a seamless way to collaborate on design proposals. They can be easily uploaded and are stored in versions. You can start a thread by clicking on the image on the exact location you would like the discussion to be focused on.

Documentation

GitLab-Figma Plugin

Our Figma plugin allows you to upload Figma frames and components to GitLab issues.

Documentation

Create new branches from issues

In GitLab, you can quickly create a new branch from an issue on the issue tracker. It will include the issue number and title automatically, making it easy to track which branch belongs to which issue.

See how in our documentation

Allow edits from upstream maintainers in a fork

When a user opens a merge request from a fork, they are given the option to allow upstream maintainers to collaborate with them on the source branch. This allows the maintainers of the upstream project to make small fixes or rebases branches before merging, reducing the back and forth of accepting community contributions.

Application performance monitoring

GitLab collects and displays performance metrics for deployed apps, leveraging Prometheus. Developers can determine the impact of a merge and keep an eye on their production systems, without leaving GitLab.

Learn more about monitoring deployed apps

Application performance alerts

GitLab allows engineers to seamlessly create service level indicator alerts and be notified of any desired events, all within the same workflow where they write their code.

Learn more about creating SLI alerts

GitLab Self-monitoring

GitLab comes out of the box enabled for Prometheus monitoring with extensive instrumentation, making it easy to ensure your GitLab deployment is responsive and healthy.

Learn more about GitLab self-monitoring

Project Level Value Stream Analytics

GitLab provides a dashboard that lets teams measure the time it takes to go from planning to monitoring. GitLab can provide this data because it has all the tools built-in: from the idea, to the CI, to code review, to deploy to production.

Learn more about Value Stream Analytics
Group Level Value Stream Analytics

GitLab provides a group dashboard that lets teams measure the time it takes for a project from planning to monitoring. GitLab can provide this data because it has an integrated CI, code review, and deployment toolset.

Learn more about Value Stream Analytics

Built-in Container Registry

GitLab's Container Registry is a secure and private registry for Docker images. It allows for easy upload and download of images from GitLab CI. It is fully integrated with Git repository management.

Documentation on Container Registry

Preview your changes with Review Apps

With GitLab CI/CD, you can create a new environment for each of your branches, speeding up your development process. Review Apps allow you to preview your branch in a live environment. Review Apps support both static and dynamic URLs.

Learn more about Review Apps

Environments Auto-stop

This feature allows users to configure an optional expiration date which can be set for review app environments.

Learn more about Environments auto-stop

New features every month

GitLab is updated with new features and improvements every month on the 22nd.

One Integrated tool

Other tools require the integration of multiple third-party tools to complete the software development lifecycle. GitLab has a completely integrated solution that covers the entire development lifecycle.

IPv6 ready

Both GitLab.com and GitLab Self-managed support IPv6.

Read the docs on configuring IPv6

AD / LDAP Integration

Sync groups, manage SSH keys, manage permissions, authentication and more. You can manage an entire GitLab instance through the LDAP / AD integration.

More information about AD / LDAP integration

Multiple LDAP / AD server support

Link multiple LDAP servers to GitLab for authentication and authorization

Advanced LDAP configuration

Access to and ability to modify source code
GitHub is publicly readable, meaning you can scan or modify the code to meet your security and development needs. The code used by most other providers is proprietary, meaning you cannot edit or view the source code.

Read the GitHub license.

Search files with fuzzy file finder

GitHub provides a way to search a file in your repository in one keystroke.

Read about the file finder in our documentation

Advanced Search

Leverage Elasticsearch for faster, more advanced code search across your entire GitHub instance.

Learn more about Advanced Search

Fast-forward merge with option to rebase

With this setting at the project level, you can ensure that no merge commits are created and all merges are fast-forwarded. When a fast-forward merge is not possible, the user is given the option to rebase.

Learn more about rebase before merge.

Remote repository push mirroring

Mirror a repository from your local server to elsewhere. Push mirroring is supported via HTTP and SSH using password authentication, and using public-key authentication with SSH.

Learn more about repository push mirroring

Disaster Recovery

Fail over in minutes to another data-center.

Learn more about Disaster Recovery

Granular user roles and flexible permissions

Manage access and permissions with five different user roles and settings for external users. Set permissions according to people’s role, rather than either read or write access to a repository. Don’t share the source code with people that only need access to the issue tracker.

Learn more about User Roles

Projects

Organize your repository into private, internal, or public projects.

Learn more about Projects

Merge Requests

Create merge requests and mention team members to review and safely merge your changes.

Learn more about merge requests

Reject unsigned commits

GitHub Premium allows you to enforce PGP signatures by rejecting unsigned commits.

Read more about enforcing push rules
Verified Committer

Verify that a push only contains commits by the same user performing the push.

In development for GitLab. Follow this link for more information.

Based on Git, a distributed VCS

Git, on which GitLab is built, is fully distributed. Every user has a complete copy of the repository, allowing for much faster access to history, easier branching and an overall better experience.

Get started with Git

Protected branches

Granular permissions for branches you want to protect.

Read about protected branches

Web IDE

Contribute to projects faster by using the Web IDE to avoid context switching in your local development environment. The Web IDE is integrated with merge requests and GitLab CI so that you can resolve feedback, fix failing tests and preview changes live with client-side evaluation without leaving the Web IDE.

Learn more about the Web IDE

Live Preview in the Web IDE

Preview changes as you make them to your JavaScript and static HTML projects with Live Preview in the Web IDE.

Learn more about the Web IDE

Web Terminal for Web IDE

Interact with your code in a Web Terminal in the Web IDE to inspect API responses, experiment in a REPL, or compile your code.

Learn more about the Web IDE Web Terminal

File Syncing to Web Terminal

Changes made in the Web IDE will now be synced to the Web Terminal. User changes made in the Web IDE can now be tested within the Web Terminal before committing them to the project.

Learn more about File Syncing to Web Terminal

EditorConfig in the Web IDE

The Web IDE supports the use of .editorconfig files in projects for standardizing coding style of all users working on the project. This helps to easily keep consistency and quality throughout the project.

Learn more about configuring the Web IDE

Paste Images in Markdown in the Web IDE

When editing Markdown files in the Web IDE you can now paste images into the content so that they’ll be automatically uploaded and referenced in the content.

Learn more about Markdown editing in the Web IDE

Real-time feedback for .gitlab-ci.yml in Web IDE
To make it easier to configure your GitLab CI pipeline, the Web IDE now provides real-time linting and completion when editing .gitlab-ci.yml files.

Learn more about .gitlab-ci.yml editing feedback in the Web IDE

**Inline commenting and discussion resolution**

Code or test reviews is faster and more effective with inline comments in merge requests. Leave comments and resolve discussions on specific lines of code. In GitLab, Merge Request inline comments are interpreted as a discussion and can be left on any line, changed or unchanged. You can configure your project to only accept merge requests when all discussions are resolved.

Learn more about resolving discussions

**Cherry-picking changes**

Cherry-pick any commit in the UI by simply clicking the Cherry-Pick button in a merged merge request or a specific commit.

Learn more about cherry picking merge requests

**Activity Stream**

View a list of the latest commits, merges, comments, and team members on your project.

Learn more about the Activity Stream

**GPG Signed Commits**

Sign commits and prove that a commit was performed by a certain user.

Read more about GPG signed commits

**Subgroups: groups within groups**

Create groups within groups to easily manage large numbers of people and projects.

Learn more about Subgroups

**Object storage for LFS**

LFS files can be stored on Object Storage (Amazon S3)

Learn how to store artifacts on object storage

**Globally distributed cloning with GitLab Geo**

When development teams are spread across two or more geographical locations, but their GitLab instance is in a single location, fetching and cloning large repositories can take a long time. Built for distributed teams, GitLab Geo allows for read-only mirrors of your GitLab instance, reducing the time it takes to clone and fetch large repos and improving your collaboration process.

Learn more about GitLab Geo

**Support for Scaled GitLab**

GitLab Premium includes support for scaling GitLab services across multiple nodes to manage demands on your system and provide redundancy. GitLab has developed reference architectures so you can easily determine the optimal architecture for your needs.

Learn more about scaling GitLab

**Deploy Boards**
GitHub Premium ships with Deploy Boards offering a consolidated view of the current health and status of each CI/CD environment running on Kubernetes. The status of each pod of your latest deployment is displayed seamlessly within GitLab without the need to access Kubernetes.

Learn more about Deploy Boards

You decide when you upgrade

GitHub releases a new version each month and lets you choose when to upgrade.

Learn how to upgrade your GitLab instance

Easy upgrade process

Using our official Linux repositories or the official Docker image, upgrading GitLab is a breeze.

Learn how to upgrade your GitLab instance

Automatically close issue(s) when a merge request is merged

With GitLab, you can use specific keywords to close one or more issues as soon as a merge request is merged.

Learn more about automatically closing issues

Work in Progress merge requests (WIP)

Prevent merge requests from accidentally being accepted before they're completely ready by marking them as Work In Progress (WIP). This gives you all the code review power of merge requests, while protecting unfinished work.

Learn more about WIP MRs

Custom Git Hooks

Leverage the power of Git Hooks and chain them together to fire off custom scripts when certain actions occur on the repository. If the commit is declined or an error occurs during the Git hook check, the error message of the hook will be present in GitLab's UI. GitLab supports all types of hooks.

Learn how to use Git Hooks with GitLab

Responsive-first design

GitLab is built with a responsive-first design approach. Use it on a desktop, tablet or smartphone. It's optimized to be viewed for the best result.

Community based, users can help shape the product

GitLab has open issue trackers for almost all of its operations. From GitLab itself to infrastructure and marketing, you can help shape the product.

View all GitLab contributors

Git LFS 2.0 support

Manage large files such as audio, video and graphics files with the help of Git LFS. Git LFS 2.0 file locking support helps large teams work with binary assets and is integrated with our native file locking feature.

Learn more about Git LFS support in GitLab

Create projects with Git push
Push new projects to the desired location and a new private project will automatically be created.

Learn more about creating Projects

SAML SSO for Groups

Connect a group in GitLab to a SAML identity provider to manage authentication.

Learn more about LDAP group synchronization

View Kubernetes pod logs

The monitoring of servers, application, network and security devices via generated log files to identify errors and problems for analysis. GitLab makes it easy to view the logs of running pods in connected Kubernetes clusters. By displaying the logs directly in GitLab, developers can avoid having to manage console tools or jump to a different interface.

Learn more about viewing Kubernetes pod logs

Restrict access by IP address

Restrict access at the group level to incoming traffic adhering to an IP address subnet, keeping your code secure.

Learn more about restricting access by IP address

Enforced Two-factor Authentication (2FA)

Two-factor authentication secures your account by requiring a second confirmation, in addition to your password. That second step means your account stays secure even if your password is compromised. The ability to enforce 2FA provides further security by making sure all users are using it.

Learn more about Enforced GitLab 2FA

Git protocol v2 support

Git’s wire protocol defines how clones, fetches and pushes are communicated between the client and server. Git protocol v2 improves performance of fetch commands and enables future protocol improvements.

Learn more about Git protocol v2

Works with multiple repository types

Supports more than one repository type, such as Git, Subversion, Perforce, CVS, Mercurial.

Learn about migrating from other SCMs

Partial Clone

Partial Clone is an optimization for very large repositories.

Learn more about Partial Clone

S/MIME Signed Commits

Sign commits and prove that a commit was performed by a certain user.

Read more about S/MIME signed commits