



GitLab

DevOps Metrics with GitLab



- What are DevOps Research and Assessment (DORA) metrics
- Why are they important
- Where can you find them in GitLab
- Other metrics available in GitLab

“I have great respect for the past. **If you don't know where you've come from, you don't know where you're going.** I have respect for the past, but I'm a person of the moment. I'm here, and I do my best to be completely centered at the place I'm at, then I go forward to the next place.”

- Maya Angelou

Is it true, is it true?



People, Process, Technology

Ask yourselves... Do you have:

Great Developers?

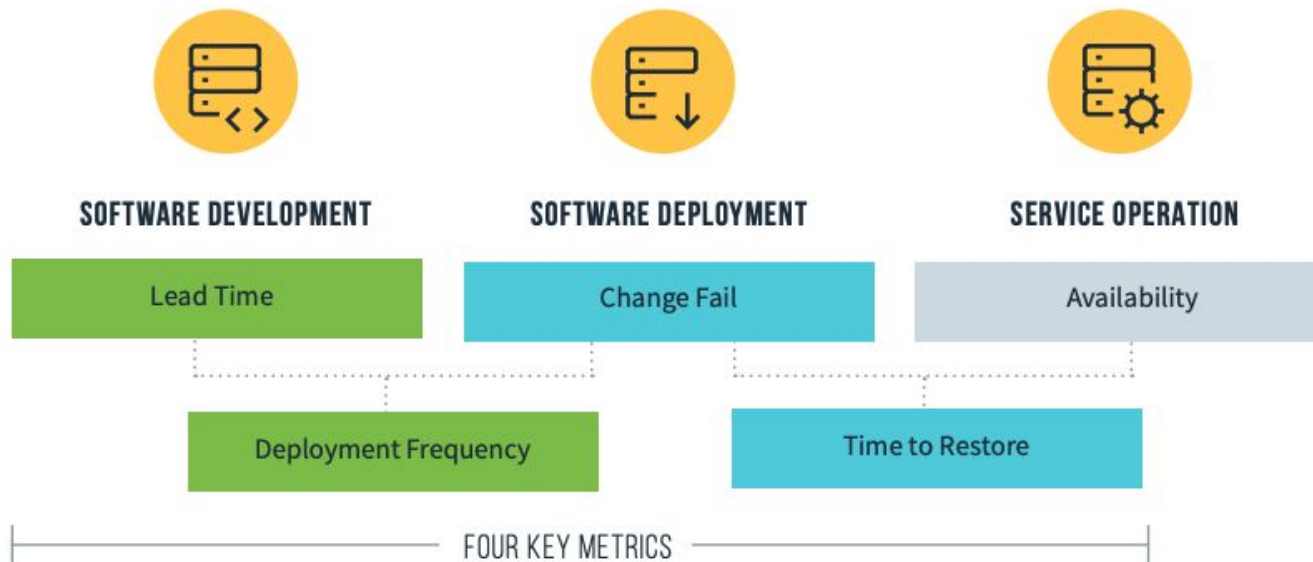
Simple Process?

Technology that works and doesn't need much to keep it that way?



DORA's research into what makes high performing technology teams is considered foundational in the DevOps space

PERFORMANCE METRICS



Where do we start?



Optimize for Performance

- Psychological safety
- Change management
- Cloud
- Technical practices
- Disaster Recovery Testing

Optimize for Productivity

- Psychological safety
- Useful easy-to-use tools
- Internal search
- External search
- Technical debt

Start by identifying your goal...



Psychological Safety



Tool: Foster psychological safety

 **YouTube** Building a psychologically safe workplace | Amy Edmondson | TEDxHGSE

re:Work

THE WORK ISSUE

What Google Learned From Its Quest to Build the Perfect Team

New research reveals surprising truths about why some work groups thrive and others falter.

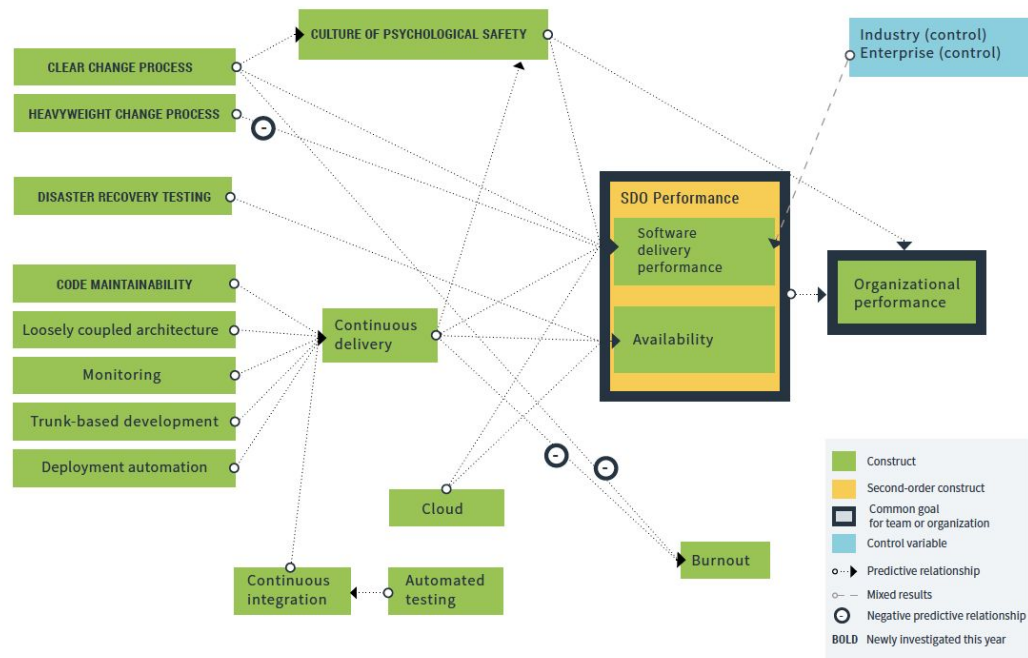


re:Work



SOFTWARE DELIVERY & OPERATIONAL PERFORMANCE

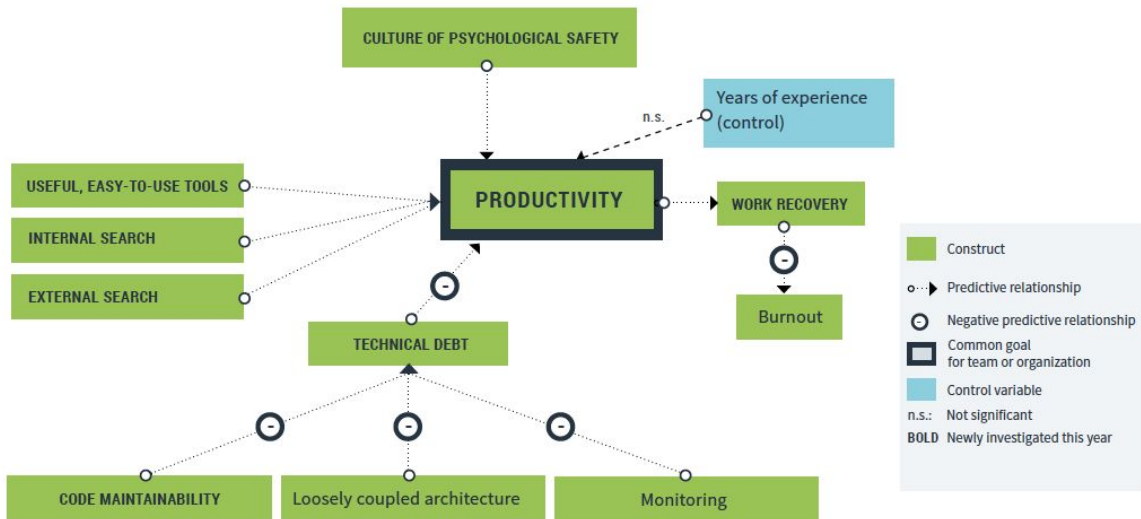
- Culture
- Clear change process
- Continuous delivery
- Cloud
- Then focus on those that are your biggest constraints.





Productivity is the ability to get complex, time-consuming tasks completed with minimal distractions and interruptions. Many of us describe this as getting into a good work flow or rhythm.

- Easy to use tools
- Manage technical debt
- Work recovery



Identify what to measure



DORA recommends:

- **Lead Time** - how long does it take to go from code committed to code successfully running in production)
- **Deployment Frequency** - how often does your organization deploy code to production or release it to end users?
- **Change Failure Rate** - what percentage of changes to production or released to users result in degraded service and subsequently require remediation
- **Time to Restore Service** - how long does it generally take to restore service when a service incident or a defect that impacts users occurs

GitLab additionally recommends:

- **Security** - how much security risk is taken on for each of these projects



Where to find these things in GitLab

Customizable Value Stream Analytics - “Lead Time”



What's in it?

- Identifies bottlenecks in development process
- Customize stages aligned with your KPIs

Helps customers answer

- How long are we taking from Idea to Production (Cycle Time)
- Where are the bottlenecks in my process?

How to navigate?

- Project: Analytics -> Value Stream Analytics

The screenshot displays the Value Stream Analytics interface. On the left, a table lists stages and their median lead times. On the right, a 'New stage' form allows adding a new stage with a name, start event, and stop event. Below the form, a smaller table shows the 'Issue start to finish' lead time for a specific issue.

Stage	Median
Issue	1 minute
Plan	Not enough data
Code	Not enough data
Test	Not enough data
Review	Not enough data
Staging	Not enough data
Total	Not enough data
In Development	25 days

New stage

Name: In Review

Start event: Issue label was added

Start event label: In review

Stop event: Issue closed

Add stage

Stage	Median	Issue start to finish
Plan	about 10 hours	Implement user #5 · Opened 22 c
Test	about 5 hours	
Review	3 days	
Staging	2 days	
Issue start to finish	21 days	

Add a stage

Edit stage

Remove stage

CI/CD Analytics - “Deployment Frequency”



What's in it?

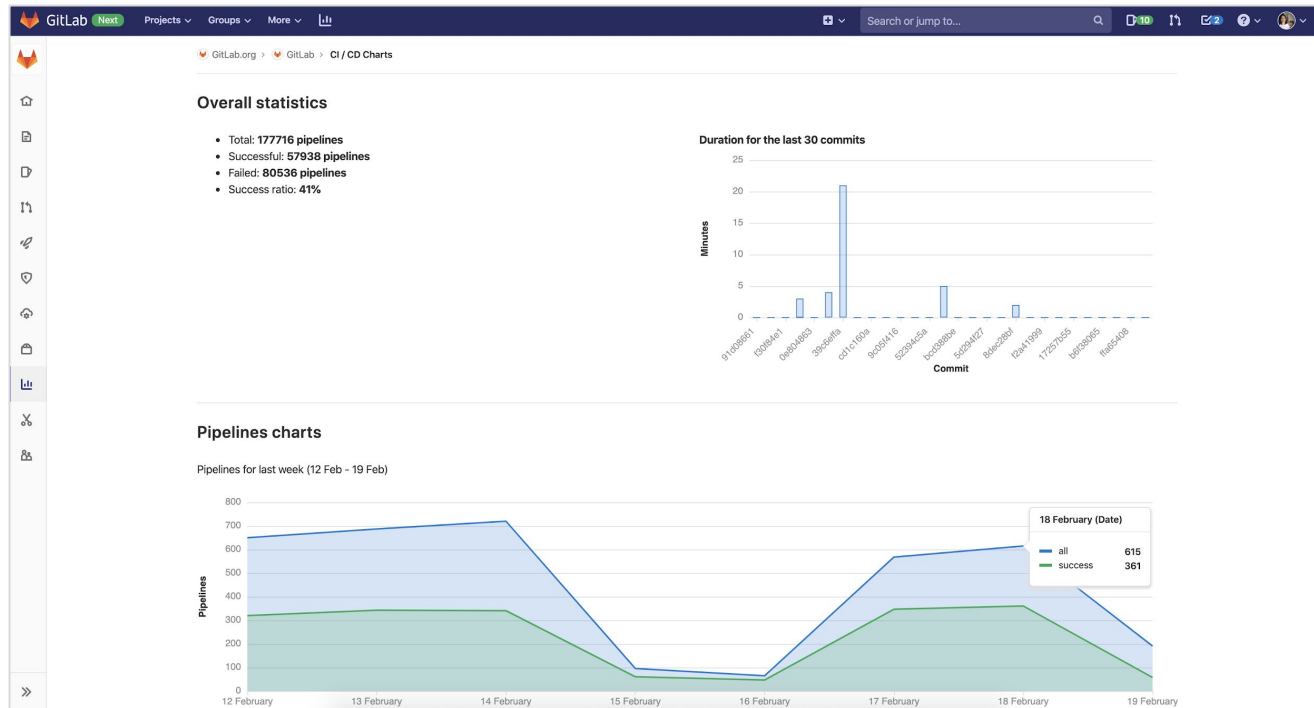
- Overall pipelines run & duration, success ratios by week, month, year

Helps customers answer

- What is my CI/CD pipeline success rate?

How to navigate?

- Project: Analytics -> CI/CD Analytics



Issue Analytics “Change Failure Rate”



What's in it?

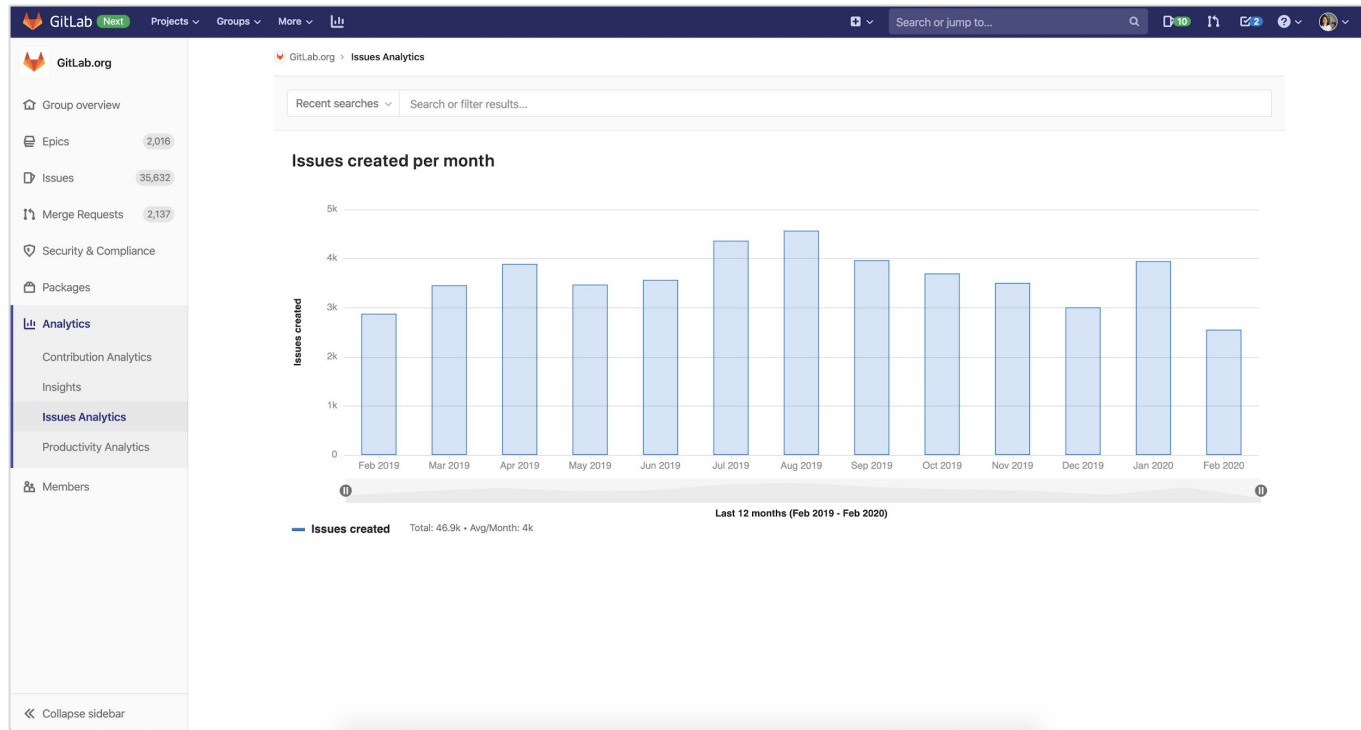
- Issues created per month by milestone, release, author etc

Helps customers answer

- What is the developer / QA productivity of my team?

How to navigate?

- Group: Analytics -> Issue Analytics



Group & Project Insights “Change Failure Rate”



What's in it?

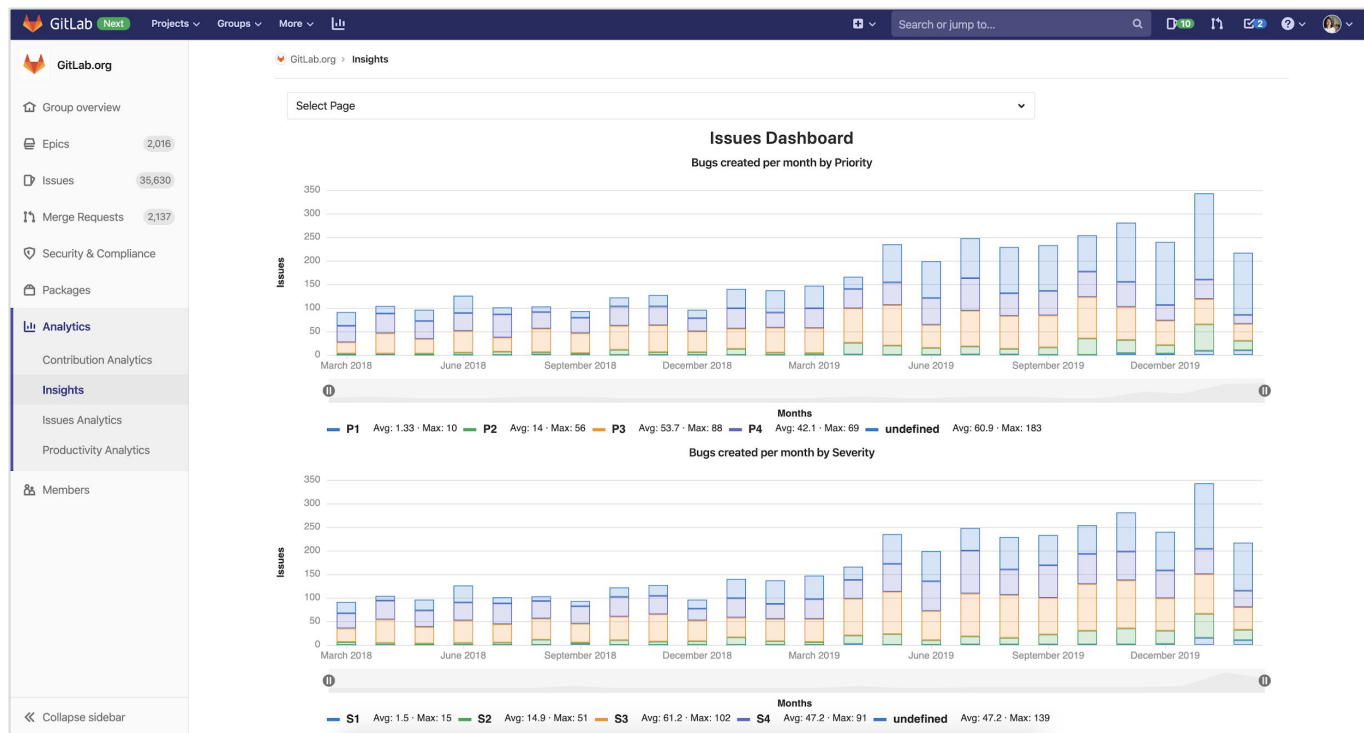
- Bugs created, merge requests, regressions, missed deadlines, track labeled issues

Helps customers answer

- What are our metrics on tracked labels? E.g., bugs, strategic issues etc

How to navigate?

- Group/Project: Analytics -> Insights



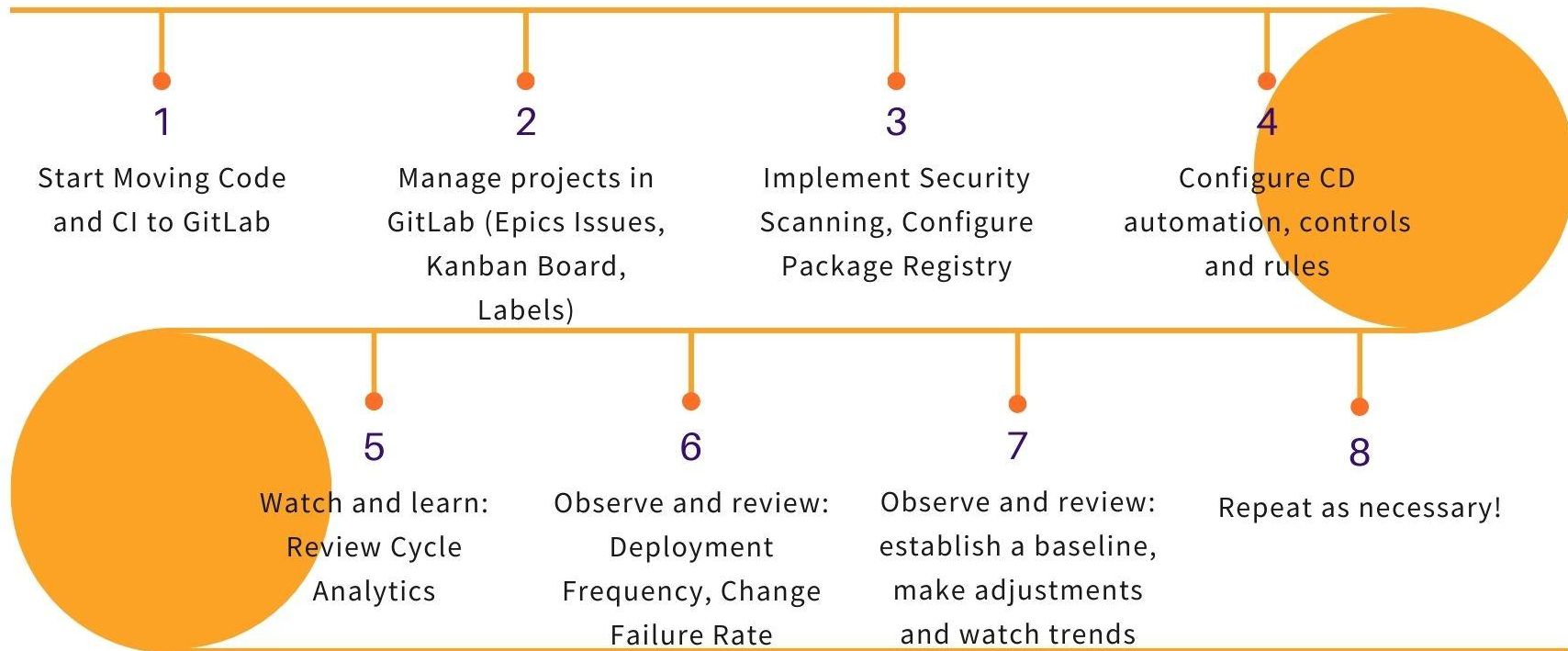
Currently Unavailable “Time to Restore Service”



But multiple options are available to derive these:

1. Incident management in GitLab could capture that and the Issue analytics can present the information.
2. Incident ticket system (assuming it is outside of GitLab can offer these metrics)

Where to begin





GitLab

But wait... There is more...



Executive Insights



What's in it?

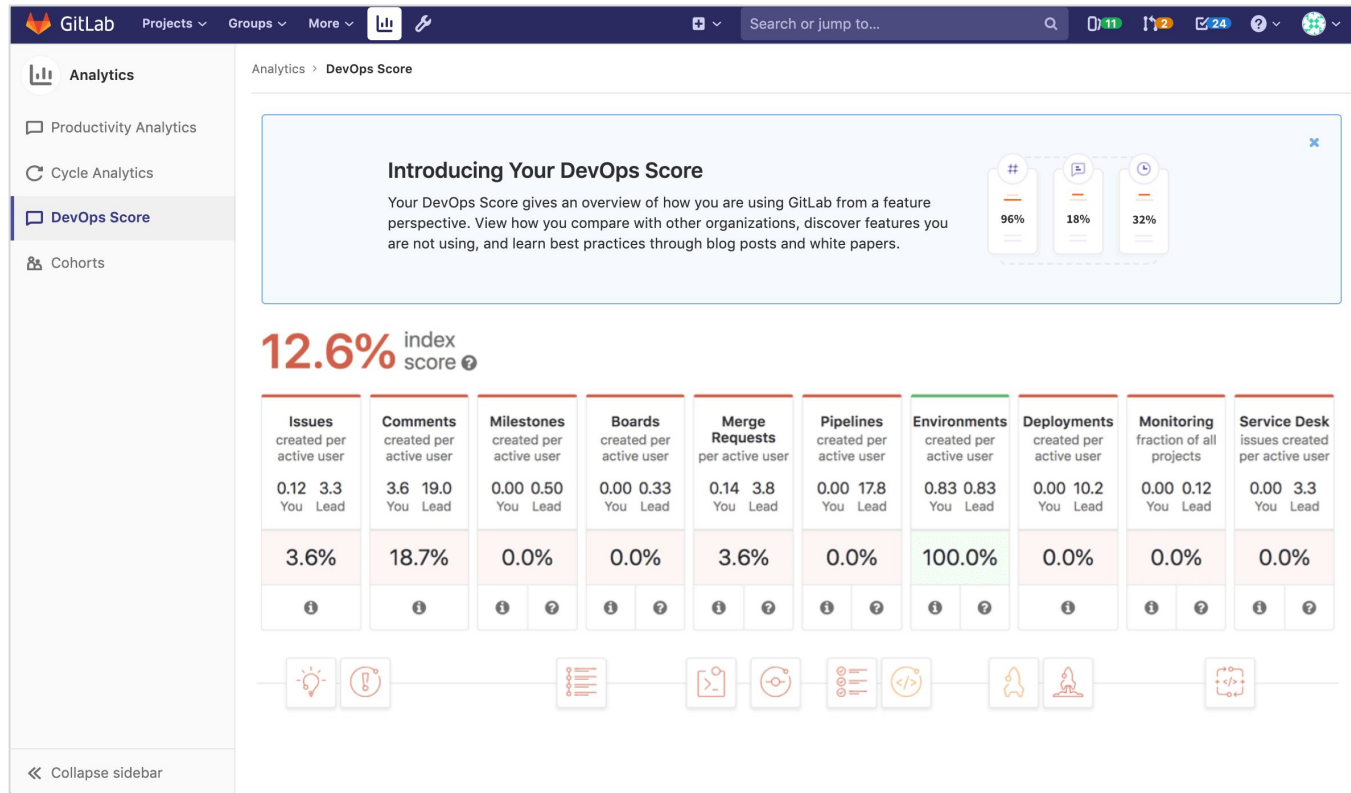
- Your organization's maturity in DevOps Adoption

Helps customers answer

- How are we doing compared to other organizations?

How to navigate?

- Analytics -> DevOps Score



Roadmaps



What's in it?

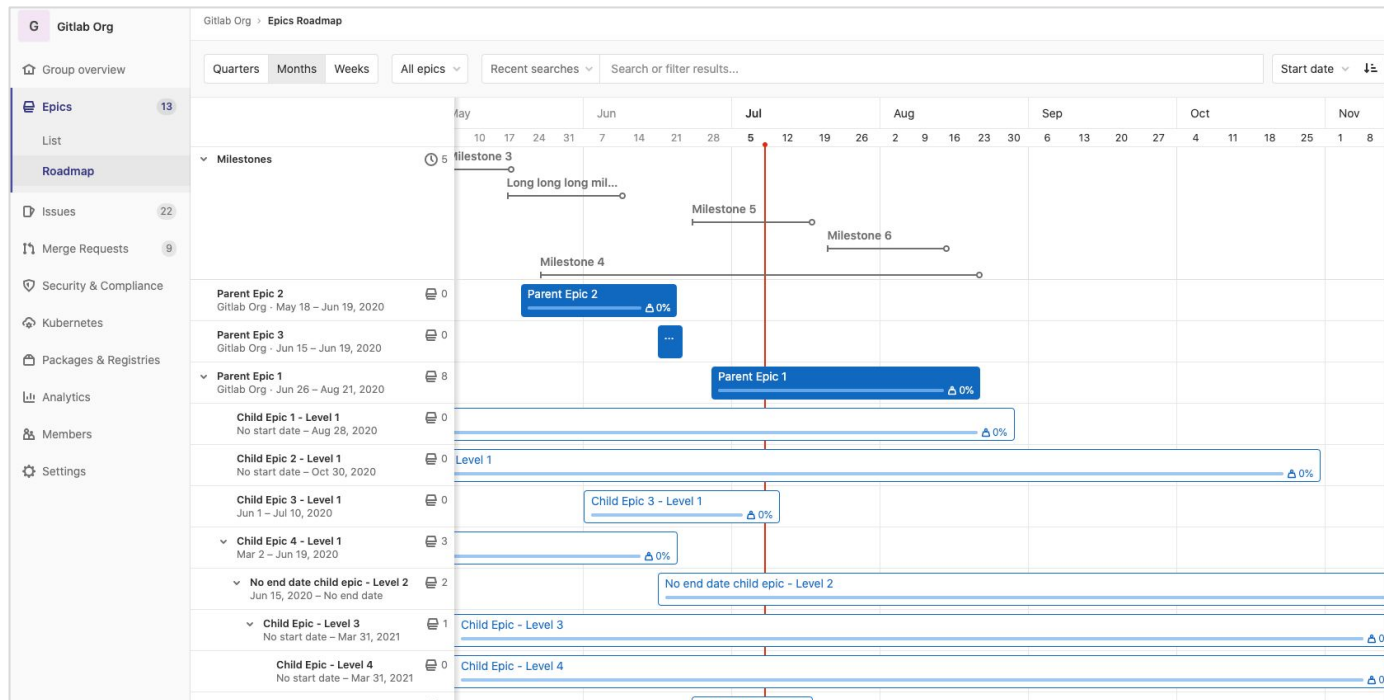
- Timelines for epics - helps with portfolio planning, tracking & execution

Helps customers answer

- Are we building the right thing at the right time?

How to navigate?

- Epics -> Roadmap





Productivity Insights



What's in it?

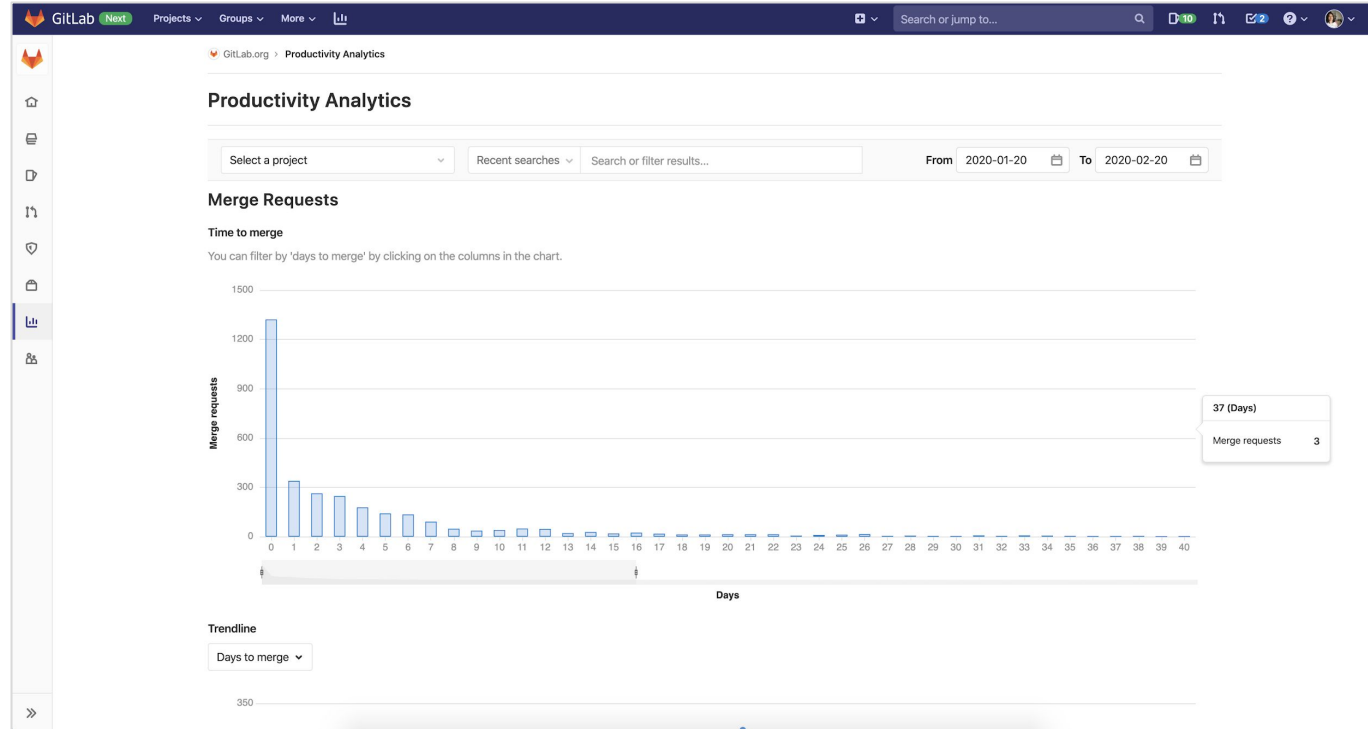
- Time to merge, commits per MR, time for first comment, lines of code per commit etc

Helps customers answer

- What is our Development Velocity at a group & project level?

How to navigate?

- Group/Project: Analytics -> Productivity Analytics



Code Review Analytics



What's in it?

- Open merge requests, review time

Helps customers answer

- What are our longest running reviews impacting cycle time?

How to navigate?

- Analytics -> Code Review Analytics

GitLab **Next** Projects Groups More **GitLab**

Search or jump to...

Code Review

Review time is defined as the time it takes from first comment until merged.

Recent searches Search or filter results...

Merge Requests in Review 1002

Merge Request	Review time	Author	Approvers	Comments	Commits	Line changes
WIP: Introduce release assets with debian r... !2234 · opened 2 years ago	967 days		—	18	1	+312 -1
EE::ProtectedRef inherits has_many and acc... !2768 · opened 2 years ago	907 days		—	4	1	+2 -10
WIP: Use case-insensitive lookup for Kerber... !3047 · opened 2 years ago · Backlog	869 days		—	9	2	+18 -1
Add note about HTTP/2 causing issues with ... !3111 · opened 2 years ago	861 days		—	4	1	+12 -0
Secret variable settings update !5974 · opened 1 year ago · 12.9	622 days		—	77	53	+245 -114
Sanitized RelayState redirect for Group SAML !6108 · opened 1 year ago · Backlog	616 days		—	10	1	+43 -1
Perform sql matching of tags ee !3628 · opened 2 years ago	552 days		—	2	3	+114 -7



Developer Insights

Contribution Analytics



What's in it?

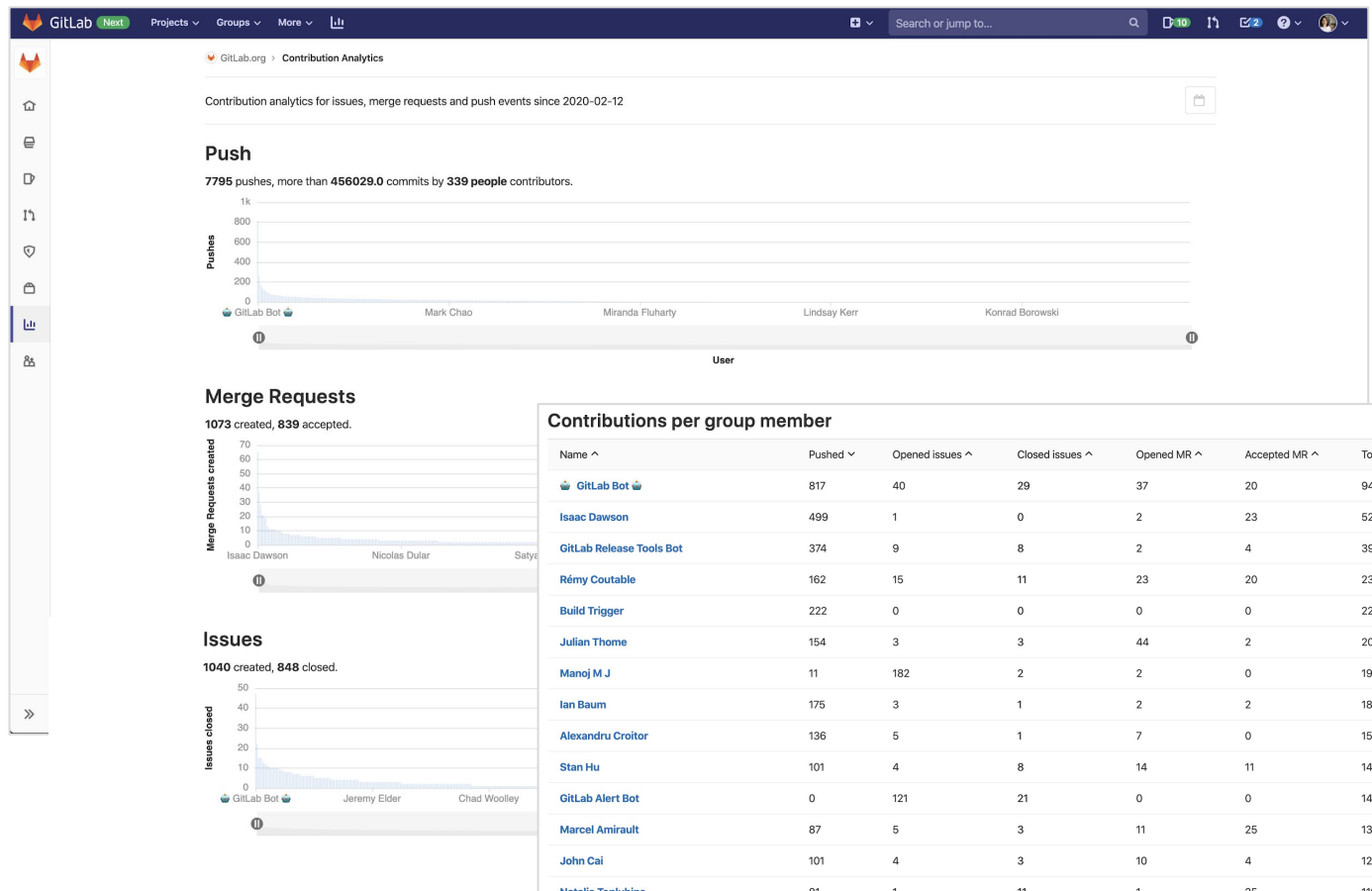
- Push events, merge requests & issues opened/closed

Helps customers answer

- At a group level, who are our top contributors and areas of improvement?

How to navigate?

- Group: Analytics -> Contribution



Contributors



What's in it?

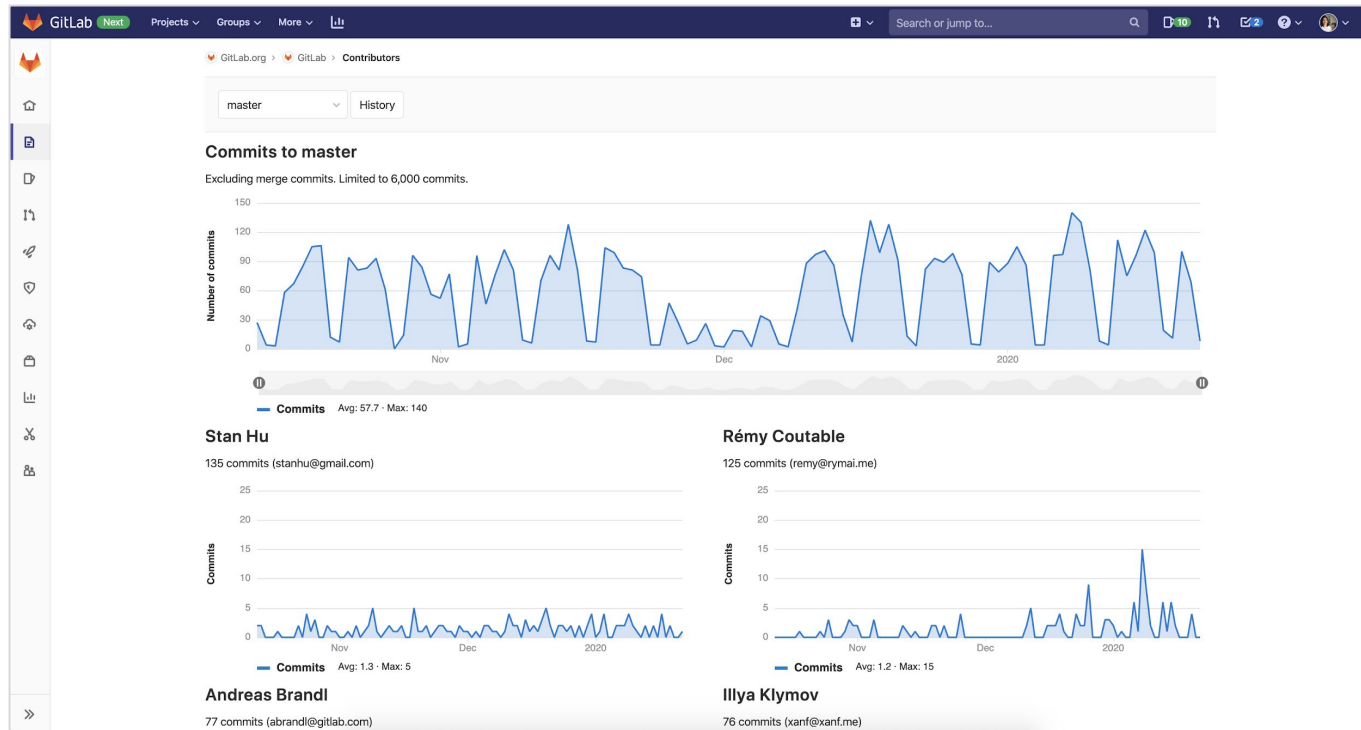
- Overall Commits to master & commits by users

Helps customers answer

- At a project level, who are our top contributors?

How to navigate?

- Project: Repository -> Contributors



Repository Analytics / Contribution Charts



What's in it?

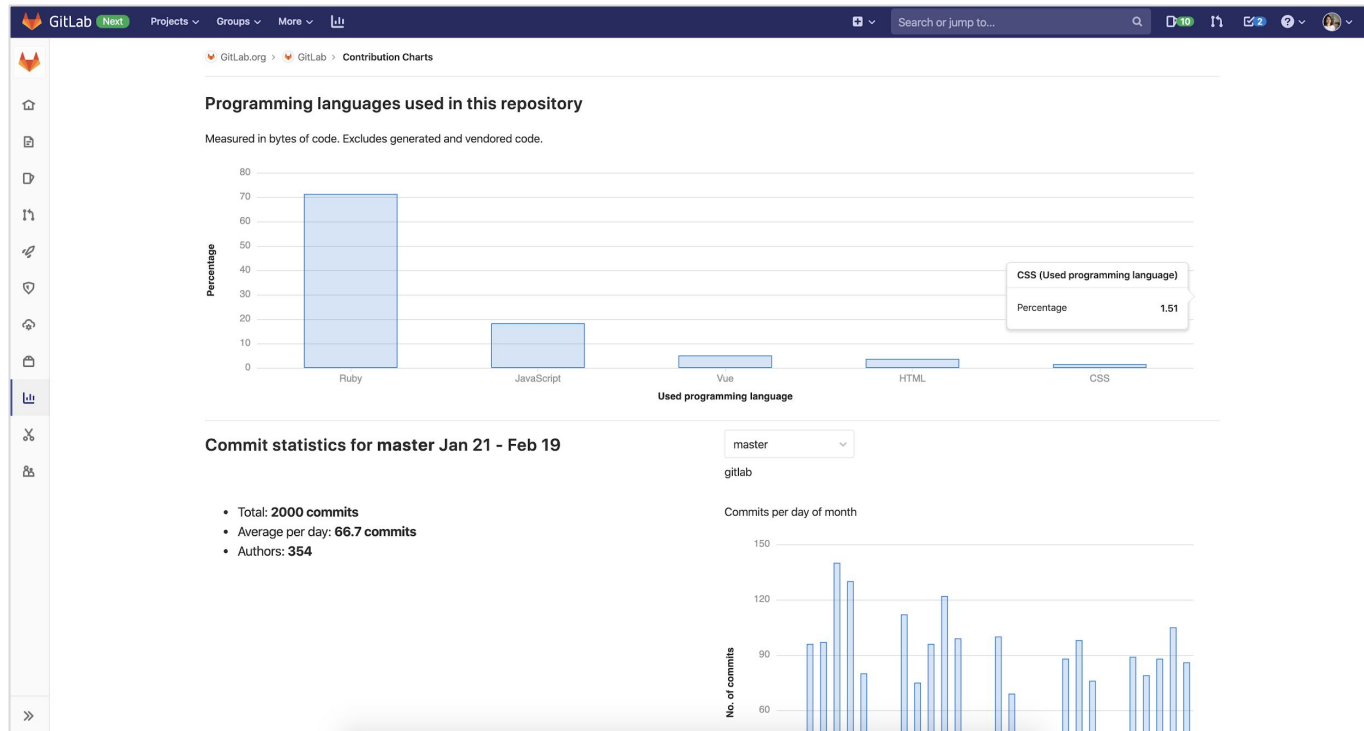
- Total commits, average per day commits, programming languages used in repository

Helps customers answer

- What programming languages do developers prefer to use?

How to navigate?

- Project: Analytics -> Repository Analytics



Code Quality Report



What's in it?

- Code Quality across the project

Helps customers answer

- What is the quality of my entire project?

How to navigate?

- Project: CI/CD -> Pipelines -> Code Quality Tab

Add new file

🕒 10 jobs for [feature-branch](#) in 36 seconds (queued for 5 minutes and 17 seconds)

🚩 latest

🔗 [0f395a0e](#) ... 📄

🔗 1 related merge request: [!1 WIP: Add new file](#)

Pipeline Jobs **10** Code Quality

⚠ Found 2258 code quality issues

- ✗ Similar blocks of code found in 2 locations. Consider refactoring. in [ee/spec/features/admin/geo/admin_geo_projects_spec.rb:152](#)
- ✗ Similar blocks of code found in 2 locations. Consider refactoring. in [ee/spec/finders/geo/lfs_object_registry_finder_spec.rb:491](#)
- ✗ Similar blocks of code found in 2 locations. Consider refactoring. in [ee/spec/finders/geo/lfs_object_registry_finder_spec.rb:512](#)
- ✗ Similar blocks of code found in 2 locations. Consider refactoring. in [ee/spec/finders/security/pipeline_vulnerabilities_finder_spec.rb:129](#)
- ✗ Similar blocks of code found in 2 locations. Consider refactoring. in [ee/spec/finders/security/pipeline_vulnerabilities_finder_spec.rb:147](#)
- ✗ Similar blocks of code found in 2 locations. Consider refactoring. in [ee/spec/lib/atlassian/jwt_spec.rb:56](#)



Operations Insights

Operations Dashboard



What's in it?

- Snapshot view of a list of projects active alerts, last commit, pipeline status, and last deployment

Helps customers answer

- What is the health of my projects?

How to navigate?

- More -> Operations

The screenshot displays the GitLab Operations Dashboard. At the top, there's a navigation bar with 'GitLab Next', 'Projects', 'Groups', and 'More'. A search bar is on the right. The dashboard title 'Operations Dashboard' is centered, with an 'Add projects' button on the right. Below the title, there's a grid of six project cards:

- GitLab.com / www-gitlab-com**: Commit `d606993a`, Merge branch 'add-link-to...', 0 Alerts, Pipeline status: `running`.
- GitLab.org / GitLab FOSS**: Commit `c1fc5da1`, Add latest changes from g..., 17 minutes ago, 0 Alerts, Pipeline status: `passed`.
- GitLab.org / gitlab-runner**: Commit `058d0c5e`, Merge branch 'replace-wh...', 1 hour ago, 0 Alerts, Pipeline status: `passed`.
- GitLab.org / gitaly**: Commit `fde7ec76`, Merge branch 'jc-add-pra...', 2 hours ago, 0 Alerts, Pipeline status: `passed`.
- monitor / Tanuki Inc**: Commit `3e7c9d3a`, Merge branch 'use-basic-...', 1 day ago, 0 Alerts, Pipeline status: `passed`.
- monitor / monitor-sandbox**: Commit `234ad0fc`, Merge branch '671-create...', 3 weeks ago, 19 Alerts, Pipeline status: `passed`.

Environments Dashboard



What's in it?

- Environments across various projects

Helps customers answer

- Are there issues with one or more of my project environments?

How to navigate?

- More -> Environments

The screenshot shows the GitLab Environments Dashboard. At the top, there's a navigation bar with the GitLab logo, 'Next' status, and tabs for Projects, Groups, and More. A search bar is on the right. Below the navigation bar, the title 'Environments Dashboard' is centered, with an 'Add projects' button on the right. A note states: 'This dashboard displays a maximum of 7 projects and 3 environments per project. [Read more.](#)'

The dashboard is divided into two main sections. The first section is for the 'monitor' project, showing a sub-project 'monitor-sandbox'. It displays a 'production' environment with a 'View app' button. Below this, a commit history is shown: 'production #411468145' by 'master' (commit 234ad0fc) 3 weeks ago, with a merge branch '671-create-...'. A green 'passed' status badge is visible.

The second section is for the 'Tanuki Inc' project. It shows two environments: 'staging' and 'production'. The 'staging' environment has a 'View app' button and shows a commit history: 'staging #350205462' by 'master' (commit fb4654d1) 3 months ago, with a merge branch 'Add new issue template'. A green 'blocked' status badge is visible. The 'production' environment also has a 'View app' button and shows a commit history: 'production #440437807' by 'master' (commit 3e7c9d3a) 1 day ago, with a merge branch 'use-basic-r...'. A green 'passed' status badge is visible.

Environments



What's in it?

- Single project environment details, ability to deploy, roll-back, monitor, stop applications in specific environment, access pod logs, web terminal & deployment history

Helps customers answer

- What is the status of my project environments?

How to navigate?

- Operations-> Environments

The screenshot shows the GitLab web interface for the 'review' environment. The top navigation bar includes 'GitLab Next', 'Projects', 'Groups', and 'More'. The breadcrumb trail is 'GitLab.org > monitor > Tanuki Inc > Environments'. The main content area displays the 'review' environment with a 'New environment' button. Below the environment name, there are tabs for 'Available' (9) and 'Stopped' (4). The environment details show a deployment for 'production' with a commit hash of '3e7c9d3a' and a job ID of 'production #440437807'. The deployment status is '100% Complete' with a 'Succeeded' status. The deployment history shows a previous deployment for 'staging' with a commit hash of 'fb4654d1' and a job ID of 'staging #350205462', which is also '100% Complete' and 'Succeeded'. The interface includes a sidebar with navigation icons and a top search bar.



Security Insights
DevSECOps

Project Security Dashboard



What's in it?

- Overview of security vulnerabilities by criticality in the project - access static, dynamic application security, dependency & container scanning results

Helps customers answer

- How secure is my application?

How to navigate?

- Security & Compliance> Vulnerability Report

The screenshot displays the GitLab interface for a project named 'Tanuki Corp - Spring App'. The left sidebar shows the 'Security & Compliance' section with the 'Vulnerability Report' selected. The main content area shows the 'Vulnerability Report' for the 'Spring' branch. It includes a summary of vulnerabilities by severity: Critical (25), High (51), Medium (127), Low (131), Info (0), and Unknown (0). Below this is a table of detected vulnerabilities with columns for Status, Severity, Description, Identifier, Scanner, and Activity.

Status	Severity	Description	Identifier	Scanner	Activity
Detected	Critical	Exposure of Resource to Wrong Sphere in org.apache.tomcat.embed/tomcat-embed-core pom.xml	CVE-2017-5648 + 1 more	Dependency Scanning	1
Detected	Critical	Information Exposure in org.apache.tomcat.embed/tomcat-embed-core pom.xml	CVE-2017-5651 + 1 more	Dependency Scanning	1
Detected	Critical	Deserialization of Untrusted Data in com.fasterxml.jackson.core/jackson-databind	CVE-2020-9546	Dependency	1

Group Compliance Dashboard



What's in it?

- Latest merge request and approvers

Helps customers answer

- Am I able to gate changes made to my environments & view a history?

How to navigate?

- Security & Compliance -> Compliance

The screenshot shows the 'Group Compliance Dashboard' for the 'Spring' subgroup. The left sidebar contains navigation links: Subgroup overview, Epics (4), Issues (72), Merge Requests (9), Security & Compliance (selected), Audit Events, Push Rules, Kubernetes, Packages & Registries, Analytics, and Wiki. The main content area is titled 'Compliance Dashboard' and includes a sub-header 'Here you will find recent merge request activity'. A table lists two merge requests:

Merge Request	Approval Status	Pipeline	Updates
Resolve "Move CSS files into static resources" tanuki-corp-sa113 - created by: Sameer Kamani	⚠️	✅	approved by: Pointy Haired Boss 26-move-css-files-into-static-resources into master merged 4 months ago
Resolve "Change the password field to fix the issue" my-tanukitech-spring!42 - created by: Sameer Kamani	⚠️	❌	approved by: Pointy Haired Boss 59-change-the-passw...-field-to-fix-the-issue into m...ter merged 8 months ago

License Compliance



What's in it?

- List of licenses used in the application

Helps customers answer

- Am I compliant to my organizational policies on licenses?

How to navigate?

- Security & Compliance> License Compliance

The screenshot shows the GitLab interface for the 'Tanuki Corp - Spring App' project. The left sidebar contains navigation links: Project overview, Repository, Issues (28), Merge Requests (3), Requirements, CI / CD, Security & Compliance (selected), Threat Monitoring, Configuration, Audit Events, and Collapse sidebar. The 'Security & Compliance' section is expanded, showing 'License Compliance' as the active tab. The main content area displays the 'License Compliance' page, which includes a breadcrumb trail: US Public Sector > Spring > Tanuki Corp - Spring App > License Compliance. Below the breadcrumb, there's a header 'License Compliance' with a help icon and a subtitle 'Displays licenses detected in the project, based on the latest successful scan • 4 hours ago'. A summary bar shows 'Detected in Project 8' and 'Policies 1'. The main table lists the detected licenses and their components:

Name	Component
unknown	hibernate (2.1.8)
GNU Lesser General Public License	logback-classic (1.1.7) and logback-core (1.1.7)
Apache Software Licenses	log4j-over-slf4j (1.7.21)
MIT License	jcl-over-slf4j (1.7.21), jul-to-slf4j (1.7.21), and 1 more
GNU Lesser General Public License v2.1 only	javassist (3.20.0-GA)
Eclipse Public License 1.0	logback-classic (1.1.7) and logback-core (1.1.7)
Apache License 2.0	classmate (1.3.3), commons-dbcp (1.4), and 34 more
Mozilla Public License 1.1	javassist (3.20.0-GA)

Dependency List



What's in it?

- List of dependencies and known vulnerabilities for the latest pipeline run

Helps customers answer

- Am I vulnerable due to any of my dependencies?

How to navigate?

- Security & Compliance> Dependency List

The screenshot shows the GitLab interface for a project named 'Tanuki Corp - Spring App'. The left sidebar contains navigation links: Project overview, Repository, Issues (28), Merge Requests (3), Requirements, CI / CD, and Security & Compliance. Under Security & Compliance, there are links to Security Dashboard, Vulnerability Report, On-demand Scans, Dependency List (selected), License Compliance, Threat Monitoring, Configuration, and Audit Events. The main content area is titled 'Dependencies' and shows a table of dependencies based on the latest successful scan. The table has columns for Component, Packager, Location, and License. Each row shows a dependency with a vulnerability status indicator (e.g., '1 vulnerability detected', '10 vulnerabilities detected').

Component	Packager	Location	License
ch.qos.logback/logback-core 1.1.7	Java (Maven)	pom.xml	1 vulnerability detected
com.fasterxml.jackson.core/jackson-databind 2.8.4	Java (Maven)	pom.xml	10 vulnerabilities detected
org.apache.tomcat.embed/tomcat-embed-core 8.5.6	Java (Maven)	pom.xml	9 vulnerabilities detected
org.springframework.boot/spring-boot 1.4.2.RELEASE	Java (Maven)	pom.xml	2 vulnerabilities detected
org.apache.tomcat.embed/tomcat-embed-websocket 8.5.6	Java (Maven)	pom.xml	1 vulnerability detected
org.hibernate/hibernate-validator 5.2.4.Final	Java (Maven)	pom.xml	1 vulnerability detected
org.springframework/spring-core 4.3.2.RELEASE	Java (Maven)	pom.xml	2 vulnerabilities detected
org.springframework/spring-web 4.3.2.RELEASE	Java (Maven)	pom.xml	3 vulnerabilities detected
org.springframework/spring-webmvc 4.3.2.RELEASE	Java (Maven)	pom.xml	5 vulnerabilities detected
junit/junit 4.12	Java (Maven)	pom.xml	1 vulnerability detected
ch.qos.logback/logback-classic 1.1.7	Java (Maven)	pom.xml	
com.fasterxml.jackson.core/jackson-annotations 2.8.4	Java (Maven)	pom.xml	

Web Application Firewall (WAF) Statistics Report



What's in it?

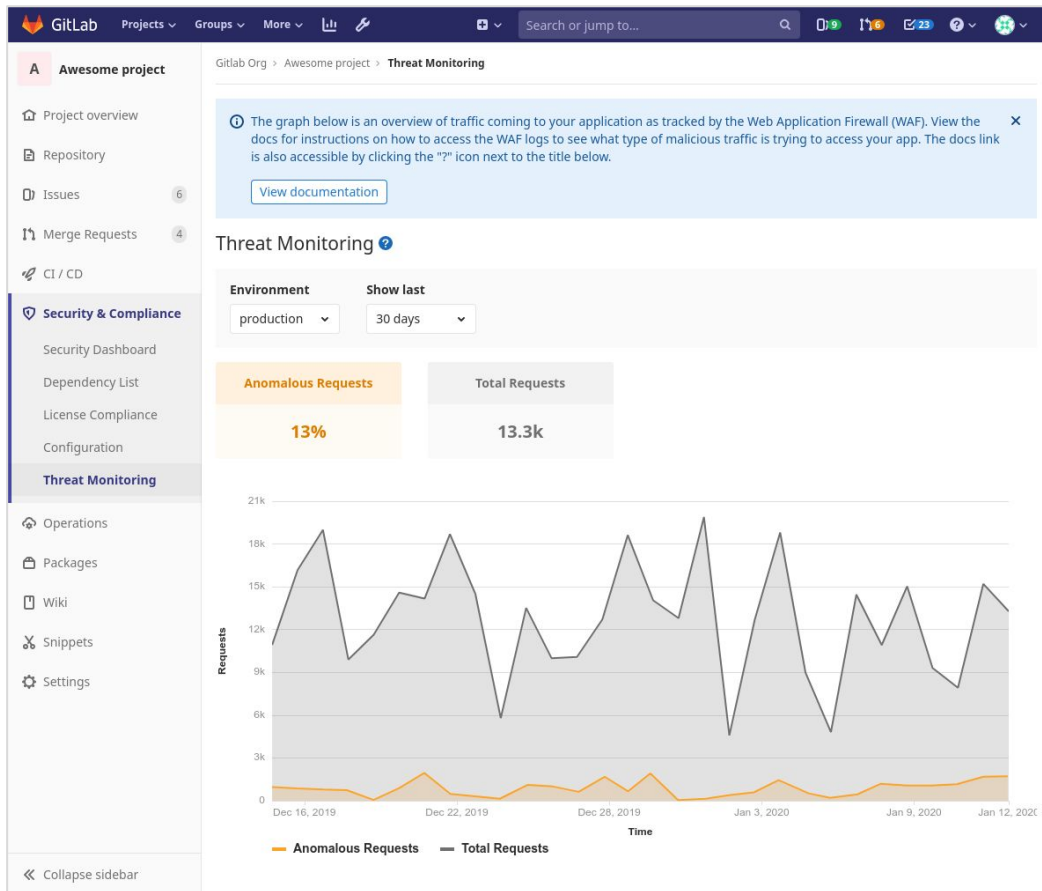
- Total and blocked traffic to determine how to configure, tune, and evaluate the Web Application Firewall

Helps customers answer

- What is my threat exposure?
Higher % of Anomalous requests imply higher threat exposure

How to navigate?

- Security & Compliance > Threat Monitoring





GitLab

Thank you!