



GLOBAL
SUBSCRIPTION
CATALYST



GS Catalyst for GitLab

GitLab Duo:
**Your AI across the entire
software delivery lifecycle**

© 2026

[Get GitLab Products Here](#)

Phones

Office : +62-21 5088 6328

Email

admin@gscatalyst.com

→ GS Catalyst – GitLab Official Partner in Indonesia



Planning and Assessment

Strategic planning, alignment of goals - business metrics baseline, deployment of resources with Professional Services Consultant



Training

Provide comprehensive training for IT teams on how to use GitLab products and integrate them into their daily operations.



Implementation

Implementation of GitLab products with certified Professional Services team



Managed Services

24/7 Managed Services help customers with proactive monitoring, incident response, and ongoing optimization

Why choose
GS Catalyst
to deliver
GitLab?

Implementation & Migration Services

Smooth implementation process and migration from existing systems.

More than 60+ Certified Engineers

Our team of certified engineers brings deep knowledge to manage your directory services

Have Strong Portfolio for Cloud Solution

Designed from the ground up for the cloud, we deliver optimal performance and scalability.

The Problem

Most environments today look like this...

74% of organizations currently using AI for software development said they wanted to consolidate their toolchain

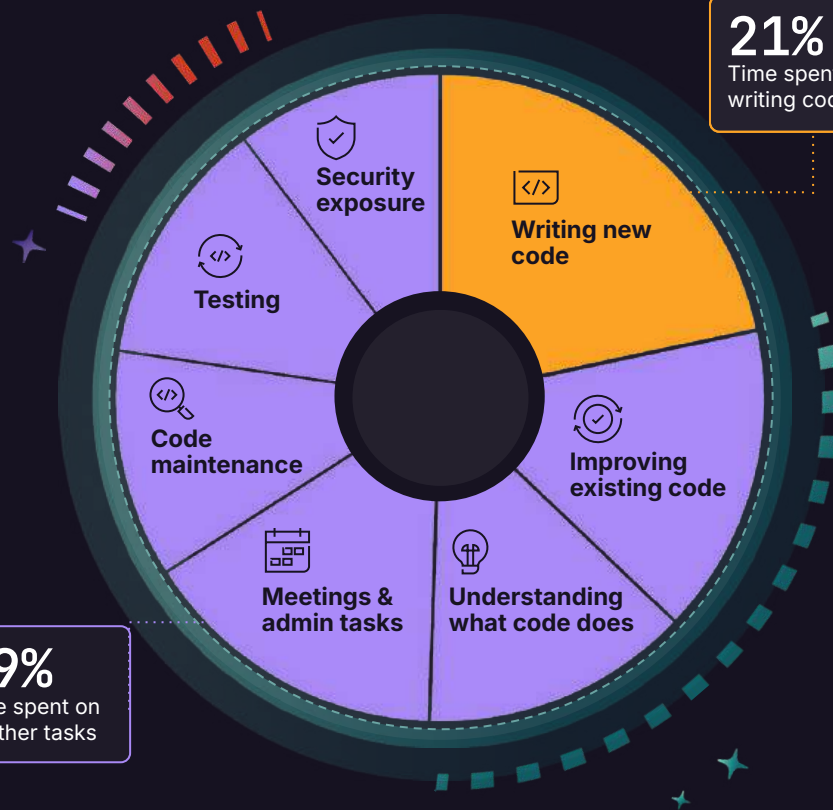


The Problem

AI adoption is only focused on coding

79%
Time spent on all other tasks

21%
Time spent writing code



The Solution



Your AI across the entire software delivery lifecycle



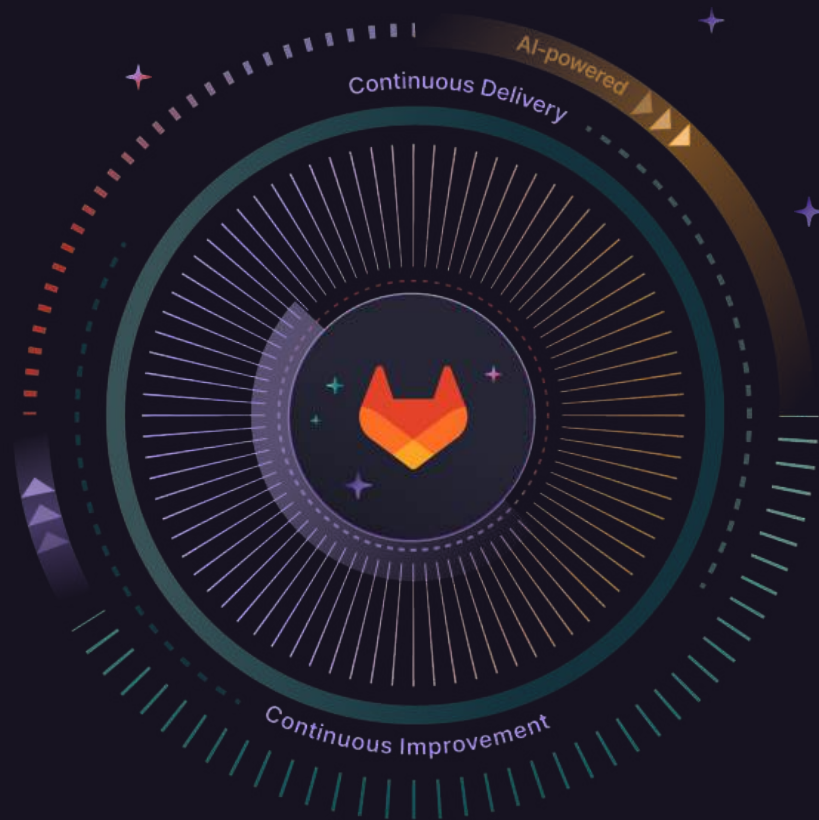
Privacy and transparency first



The right AI models for each use case



Unified data store that provides full context of your workflow

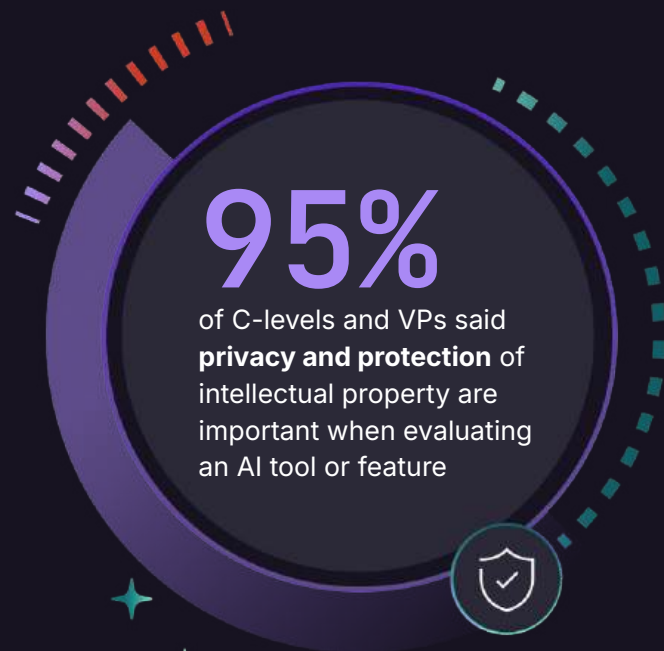


How we differentiate

Transparency and privacy first

GitLab Duo does not use your proprietary code as training data. The models we use are not trained on private GitLab data.

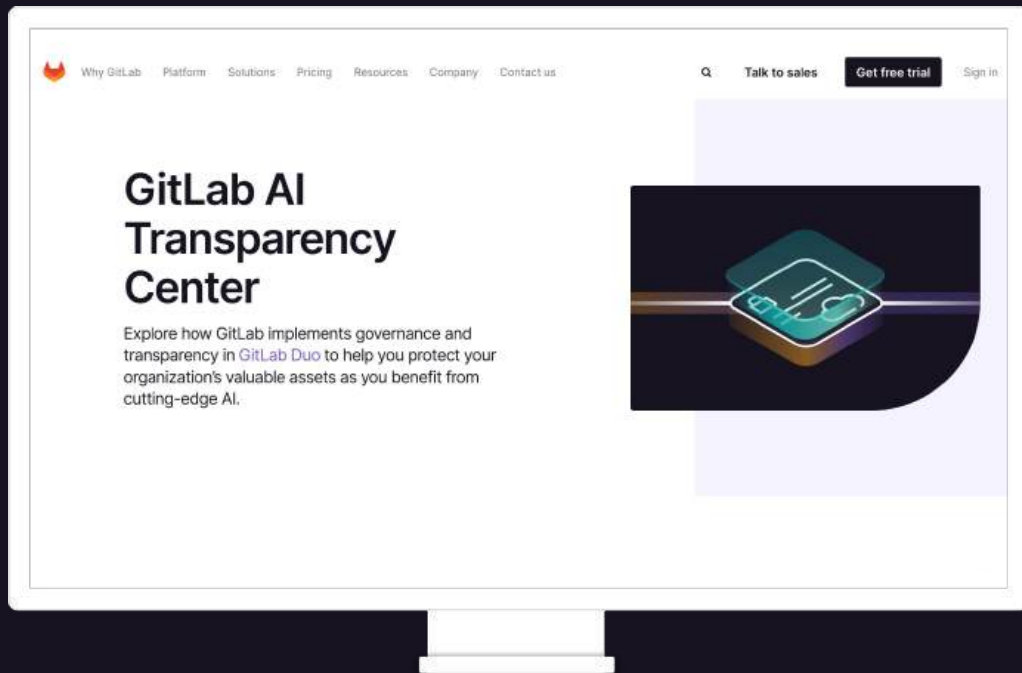
Our publicly available documentation describes all AI models used by GitLab Duo and how we're using your code base.



GitLab AI Transparency Center

Helps customers understand how GitLab upholds ethics and transparency in our AI-powered features. The AI Transparency Center includes:

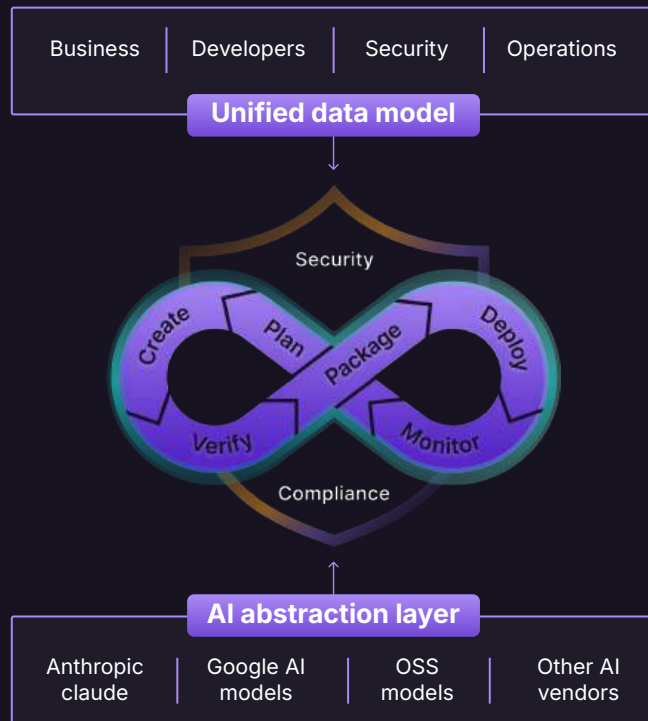
- ✓ AI Ethics for Product Development
- ✓ AI Continuity Plan
- ✓ AI Features Documentation



How we differentiate

The right AI model for each use case

A multi-model approach allows GitLab Duo to dynamically change models to ensure the highest quality result.

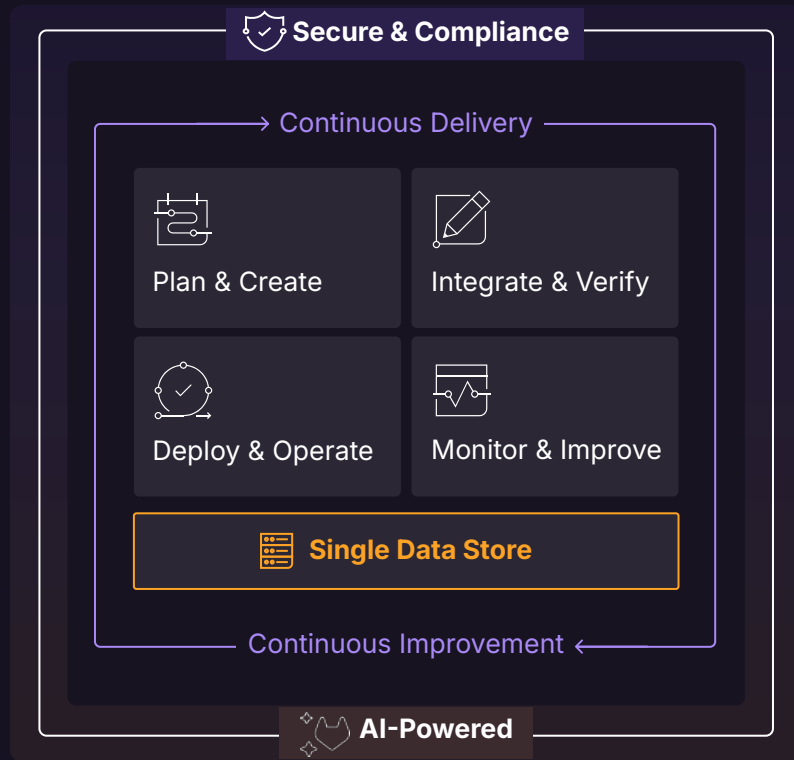


How we differentiate

Unified Data Store

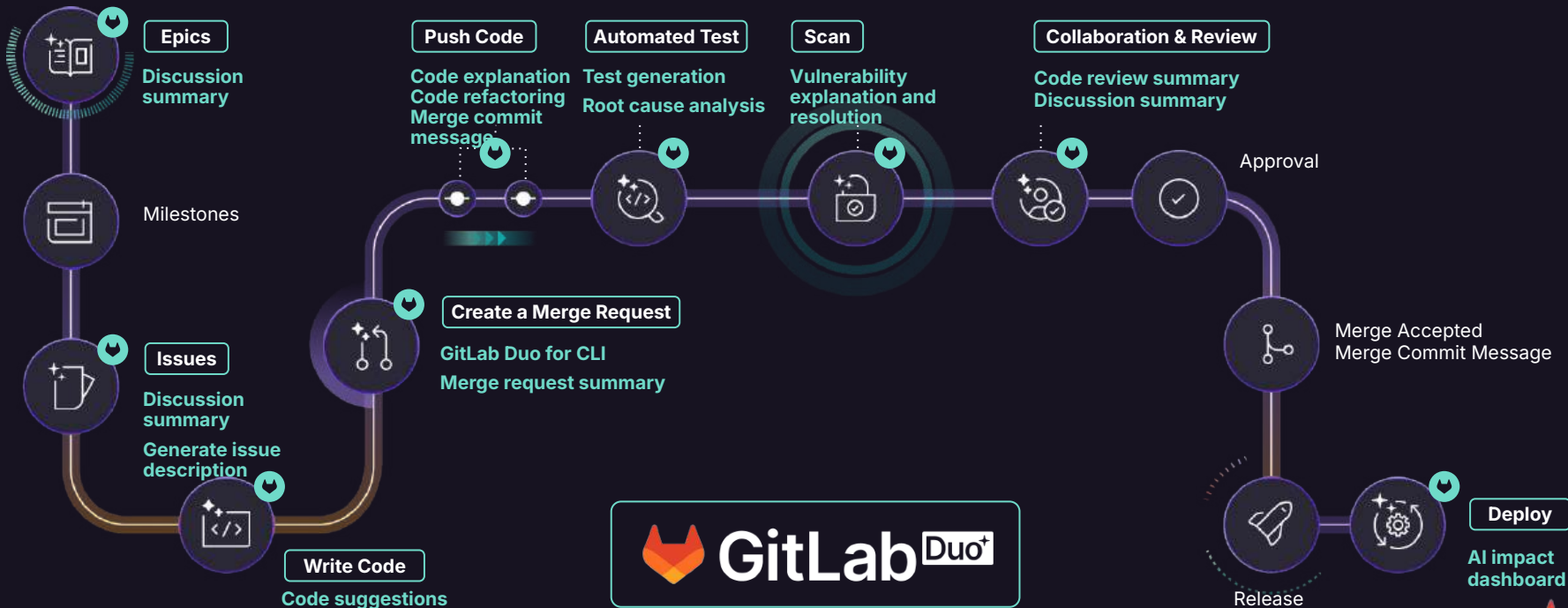
Built with a unified data store, only GitLab Duo can provide full context of a user's entire workflow.

Enabling teams to accelerate software development, improve security, and provide measurable ROI



The most comprehensive AI-powered DevSecOps platform

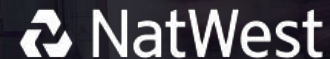
GitLab Duo Chat





GitLab Duo enables our developers to be more productive, efficient, and successful in creating secure code. We're excited to see the benefits of GitLab's AI features across the entire value chain, and even our most seasoned engineers are seeing value.

Bal Kang
Head of Digital Platform Engineering



GitLab recognized as a Leader

2024 Gartner[®] Magic Quadrant[™] for AI Code Assistants

According to Gartner, Leaders execute well against their current vision and are well positioned for tomorrow.



Source: Gartner, Magic Quadrant for AI Code Assistants, Arun Batchu, Haritha Khandabattu, Philip Walsh, Matt Brasier, August 2024 | GARTNER is a registered trademark and service mark of Gartner, Inc. and/or its affiliates in the U.S. and internationally, and MAGIC QUADRANT is a registered trademark of Gartner, Inc. and/or its affiliates and are used herein with permission. All rights reserved. | Gartner does not endorse any vendor, product or service depicted in its research publications, and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose. | This graphic was published by Gartner Inc. as part of a larger report and should be evaluated in the context of the entire document. The Gartner document is available upon request from Gartner B.V.



GitLab Duo Workflow

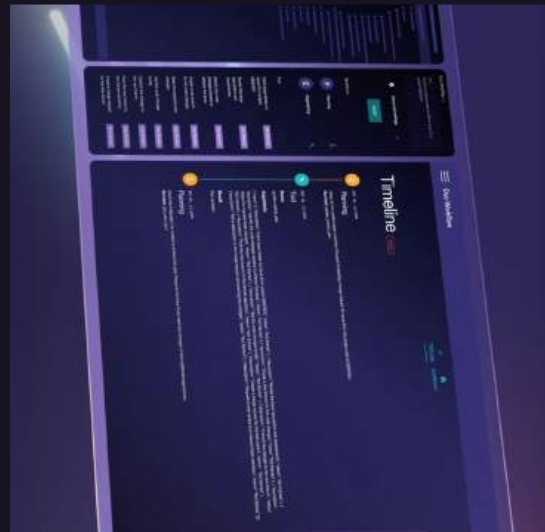
The future of AI-driven software development

Fully autonomous AI agent that proactively supports developers as they develop, deploy, and secure software.

Transitioning from reactive, prompt-based AI to proactive, autonomous AI that drives software development and delivery.



GitLab Duo Workflow video



GitLab **Duo** Enterprise



Organizational user controls

User permissions for AI capabilities



Code suggestions

- Code Generation
- Code Completion
- Available in many popular IDEs and supports 25+ programming languages



Chat

- Issue and Epic Management
- Code Explanation
- Test Generation
- Code Refactoring



Summarization & templating tools

- Discussion Summary
- Merge Request Summary
- Code Review Summary
- Merge Commit Message



Security & vulnerability tools

- Vulnerability Explanation
- Vulnerability Resolution



Advanced troubleshooting

- Root Cause Analysis



AI analytics

- AI Impact Dashboard



Personalize GitLab Duo to your organization

- Self-Hosted Model Deployment*
- Model Personalization*

** Planned. Additional terms and fees may apply.*

Now available!

\$39

per user/month

Billed annually





Your AI across the entire software delivery lifecycle



Privacy and transparency first




The right AI models for each use case




Unified data store that provides full context of your workflow




GitLab Duo⁺ Pro Pricing


 Code completion and code generation with Code Suggestions

 Organizational user controls

 Chat

 Code explanation

 Code refactorization

 Test generation

\$19
per user/month

Bills annually at \$228 USD

Available in many popular IDEs and supports 25+ programming languages.

Code Suggestions for Self-Managed customers requires GitLab 16.9 or later.



Duo Enterprise

Key Capabilities

AI impact dashboard

Track the effectiveness of AI with GitLab Duo's AI Impact Dashboard. See real-time improvements in cycle times and deployment frequencies, making it easy to evaluate your ROI in actual numbers, not just promises.

Root cause analysis

GitLab Duo Root Cause Analysis assists developers in determining a root cause and suggesting a fix for a CI/CD job log failure by analyzing the logs, allowing developers to focus on critical tasks.

Vulnerability explanation & resolution

Our integrated security capabilities ensure that AI outputs are verified by multiple checkpoints throughout the development workflow. GitLab Duo Vulnerability explanation and remediation fortify your software against vulnerabilities.

Summarization & templating tools

GitLab Duo's summarization and templating tools streamline communication within and between teams.





Appendix

Feature by feature slides

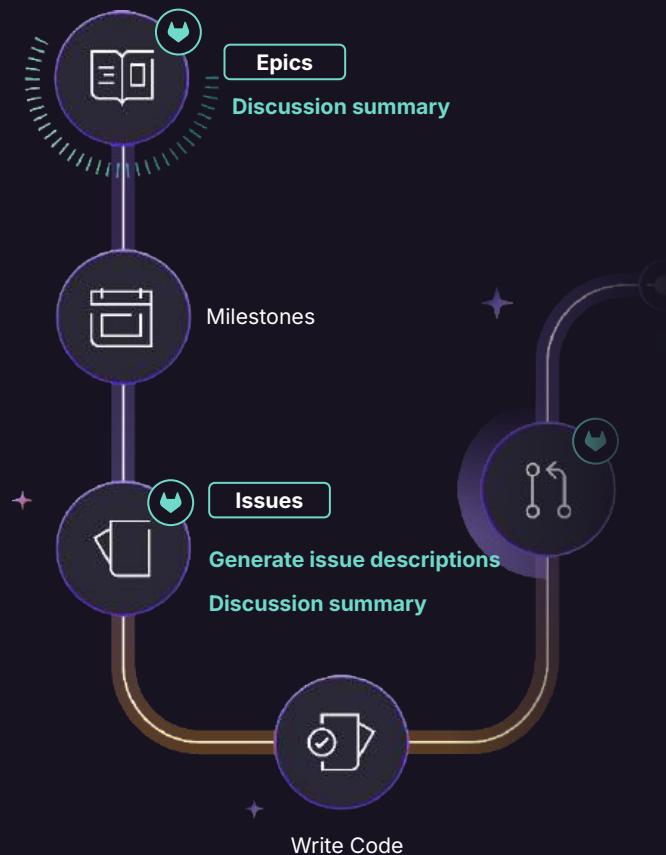
To add after slide 12



Improving collaboration and developer experience

AI-powered automation for:

- ✓ **Generate issue descriptions**
Allows users to get started faster
- ✓ **Discussion summary**
Accelerates team collaboration by onboarding people faster



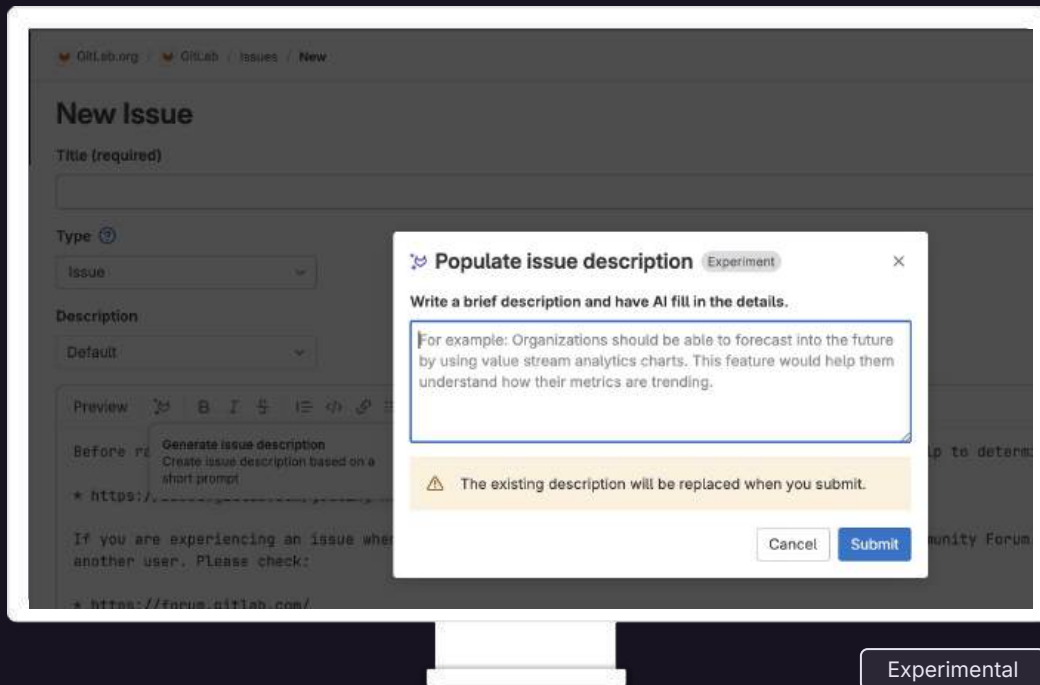
Ultimate

SaaS

Generate issue description

Quickly generate a new issue based on issue templates

- ✓ Drive consistency with issue templates
- ✓ Helps ensure complete issue descriptions to keep everyone on the same page



Ultimate

SaaS

Self Managed

Discussion summary

Summarize long issue, epic and
merge request comment threads

- ✓ Quickly get up to speed on lengthy conversations
- ✓ Get everyone on the same page and take action faster

The screenshot shows a GitLab issue page for issue #100, titled "DataSet Curation for Explain the Vulnerability (ETV)". The breadcrumb trail is: GitLab.org > Code Suggestions > Prompt Library > Issues > #100. The issue is marked as "Open". A callout box points to the "View summary" button, stating "Generates a summary of all comments". The "Activity" section shows an "AI-generated summary" for an "Experiment". The summary includes:

- State of Things**
 - The dataset has been loaded into BigQuery: unreview-poc-390200e5.llm_sec_eval.diverse_vul
 - The dataset has been loaded into the dataset labeled llm_sec_eval because it hasn't been changed to say explain_this_vulnerability yet.
- Schema**
 - The schema of the dataset is provided.
- Exploratory Data Analysis (EDA)**
 - There are 330492 records total.
 - Only 14% (48758 / 330492) of the records are missing CWE identifiers, some have more than 1. The majority though, only have 1

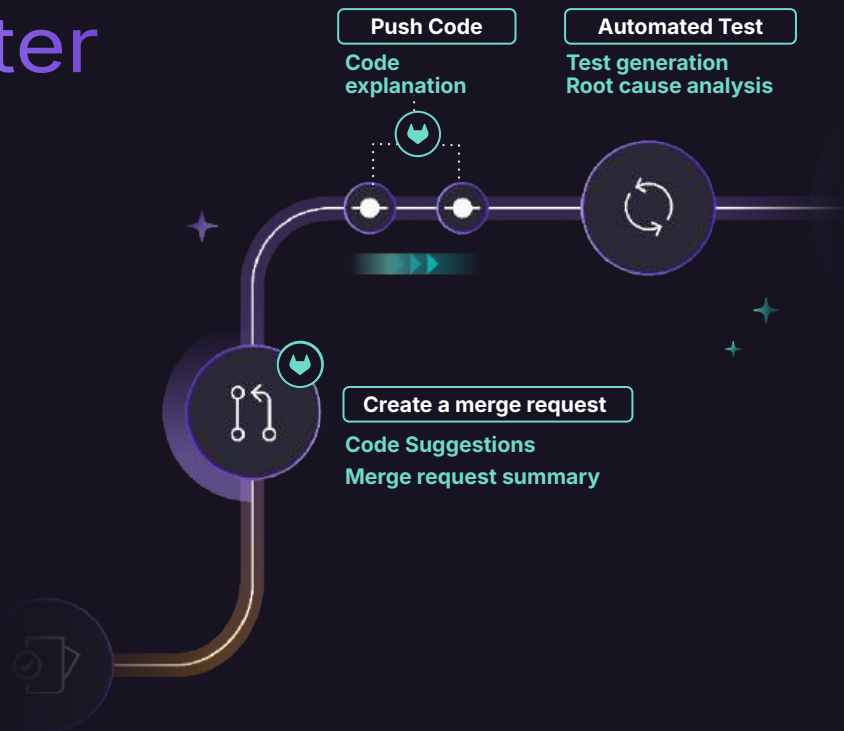




Creating better code, faster

AI-powered automation for:

- ✓ **Code Suggestions**
Accelerates writing new code or rewriting existing code
- ✓ **Code explanation**
Improve developer skills and understanding
- ✓ **Test generation**
Improving code quality before deployment
- ✓ **Root cause analysis**
Determine cause of failed CI/CD build



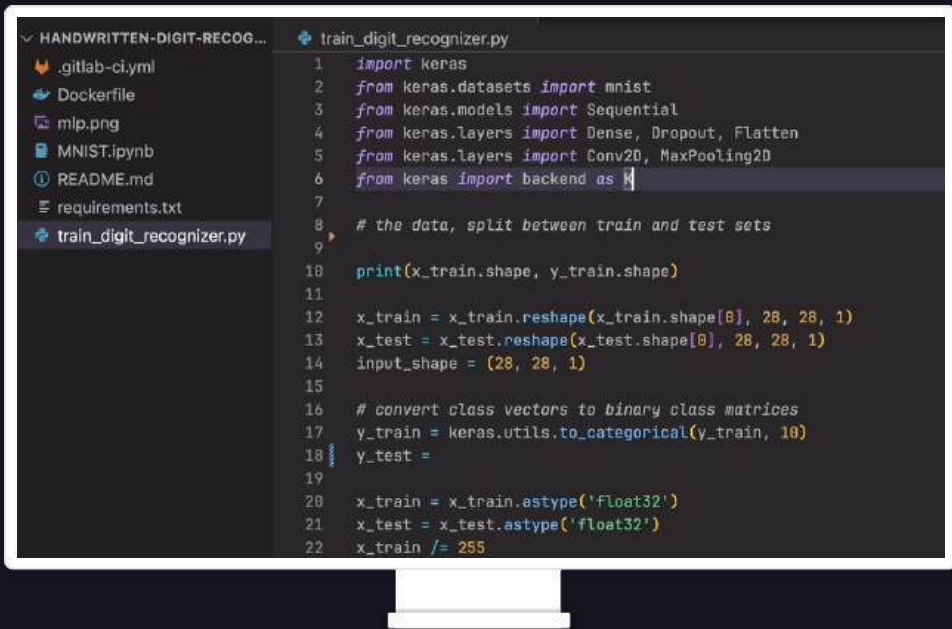
All paid tiers

SaaS

Self Managed

Code Suggestions

- ✓ Improves developer efficiency and velocity through code recommendations.
- ✓ Support for [25+ languages](#): C++, C#, C, Go, Google SQL, Java, JavaScript, Kotlin, Markdown, PHP, Python, Ruby, Rust, Scala, Swift, TypeScript, and more via configuration.
- ✓ Support for infrastructure files: Google Cloud CLI, Kubernetes Resource Model (KRM), Terraform
- ✓ IDE Support includes: VS Code, GitLab WebIDE, MS Visual Studio, JetBrains based IDEs, Neovim, Android Studio



```
✓ HANDWRITTEN-DIGIT-RECOG...  train_digit_recognizer.py
🔥 .gitlab-ci.yml
🐋 Dockerfile
🖼️ mlp.png
📄 MNIST.ipynb
📄 README.md
📄 requirements.txt
🔗 train_digit_recognizer.py
1  import keras
2  from keras.datasets import mnist
3  from keras.models import Sequential
4  from keras.layers import Dense, Dropout, Flatten
5  from keras.layers import Conv2D, MaxPooling2D
6  from keras import backend as K
7
8  # the data, split between train and test sets
9
10 print(x_train.shape, y_train.shape)
11
12 x_train = x_train.reshape(x_train.shape[0], 28, 28, 1)
13 x_test = x_test.reshape(x_test.shape[0], 28, 28, 1)
14 input_shape = (28, 28, 1)
15
16 # convert class vectors to binary class matrices
17 y_train = keras.utils.to_categorical(y_train, 10)
18 y_test =
19
20 x_train = x_train.astype('float32')
21 x_test = x_test.astype('float32')
22 x_train /= 255
```



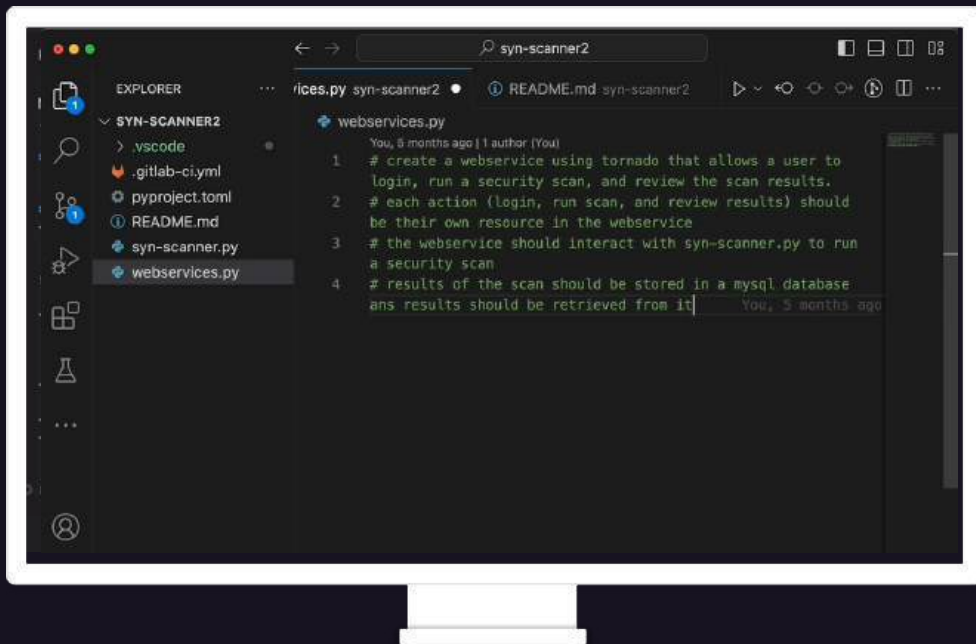
All paid tiers

SaaS

Self Managed

Repository X-Ray

- ✓ Provides additional context and improves accuracy and relevance of suggestions
- ✓ Analyzes dependency files such as Gemfile.lock, package.json, and go.mod to build additional context
- ✓ Helps tailor suggestions to match the coding patterns, styles and technologies used in the project
- ✓ Code recommendations integrate more seamlessly and follow best practices for your stack



All paid tiers

SaaS

Self Managed

Code explanation

Available in Chat

Inline code explanations

- ✓ Enables developers to quickly understand code
- ✓ Uplevels developers through continuous learning

The screenshot displays a GitLab Duo Chat window overlaid on a code editor. The code editor shows a Python file named `train_digit_recognizer.py` with 26 lines of code. The code imports Keras and uses `mnist.load_data()` to load training and testing data. It then reshapes the data and uses `keras.utils.to_categorical()` to convert the target variables `y_train` and `y_test` into one-hot encoded format. A question mark icon is placed next to line 18, which contains the `keras.utils.to_categorical(y_train, 10)` call. The chat window on the right, titled "GitLab Duo Chat", shows the AI's response to this question. The response explains that the selected code snippet is used to convert target variables into one-hot encoded format for multi-class classification. It notes that the problem is to classify handwritten digits from 0 to 9, which is a 10-class classification problem. The chat also provides the function signature `keras.utils.to_categorical(y_train, 10)` and lists the arguments: `y_train` or `y_test` (target variables) and `10` (number of classes). A final note states that the function converts class labels into a one-hot encoded format where each label is represented as a binary vector of length 10.

```
1 import keras
2 from keras.datasets import mnist
3 from keras.models import Sequential
4 from keras.layers import Dense, Dropout, Flatten
5 from keras.layers import Conv2D, MaxPooling2D
6 from keras import backend as K
7
8 # the data, split between train and test sets
9 (x_train, y_train), (x_test, y_test) = mnist.load_data()
10
11 print(x_train.shape, y_train.shape)
12
13 x_train = x_train.reshape(x_train.shape[0], 28, 28, 1)
14 x_test = x_test.reshape(x_test.shape[0], 28, 28, 1)
15 input_shape = (28, 28, 1)
16
17 # convert class vectors to binary class matrices
18 y_train = keras.utils.to_categorical(y_train, 10)
19 y_test = keras.utils.to_categorical(y_test, 10)
20
21 x_train = x_train.astype('float32')
22 x_test = x_test.astype('float32')
23 x_train /= 255
24 x_test /= 255
25 print('x_train shape:', x_train.shape)
26 print(x_train.shape[0], 'train samples')
```

GitLab Duo Chat

Responses generated by AI

The selected code snippet is used to convert the target variables `y_train` and `y_test` into a one-hot encoded format, which is required for multi-class classification problems. In this case, the problem is to classify handwritten digits from 0 to 9, which is a 10-class classification problem.

```
y_train = keras.utils.to_categorical(y_train, 10)
y_test = keras.utils.to_categorical(y_test, 10)
```

The `keras.utils.to_categorical` function takes two arguments:

- `y_train` or `y_test`: These are the target variables containing the class labels (in this case, digits from 0 to 9).
- `10`: This is the number of classes in the classification problem (0 to 9, inclusive).

The function converts the class labels into a one-hot encoded format, where each label is represented as a binary vector of length 10.



All paid tiers

SaaS

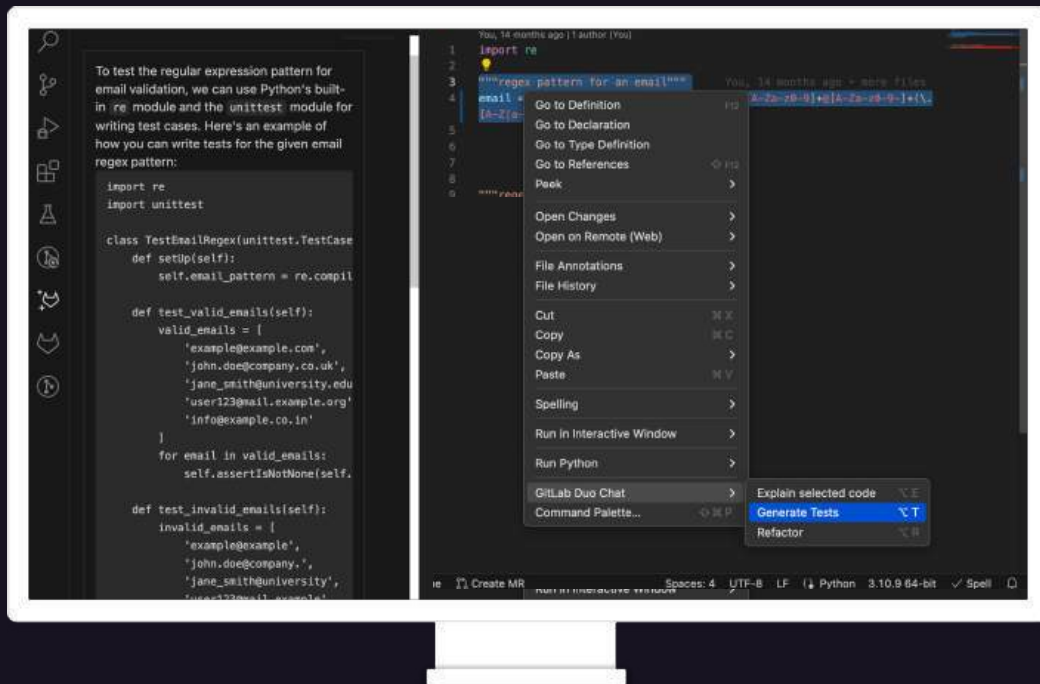
Self Managed

Test generation

Available in Chat

Quickly add unit tests to merge requests

- ✓ Automates repetitive task for developers and helps catch bugs early
- ✓ Improves test coverage early in the development cycle



Ultimate

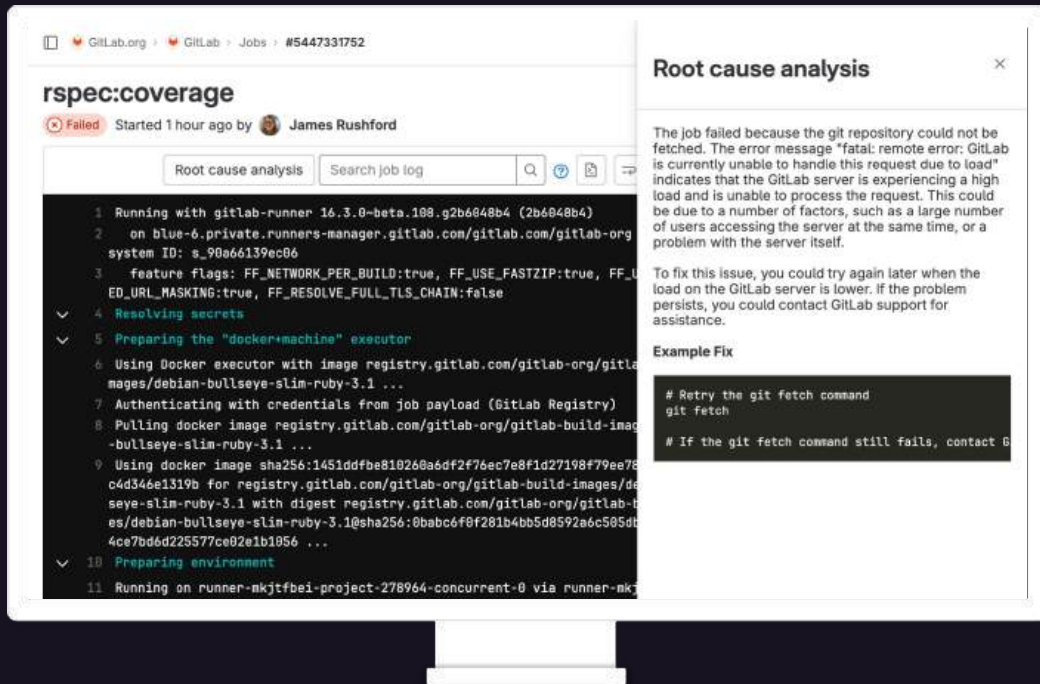
SaaS

Self Managed

Root cause analysis

Quickly determine the cause of a failed CI/CD build

- ✓ Removes the manual task of sorting through lengthy job logs
- ✓ Suggests potential ways to fix failed jobs, reducing debugging time



GitLab.org > GitLab > Jobs > #5447331752

rspec:coverage

Failed Started 1 hour ago by James Rushford

Root cause analysis Search job log

```
1 Running with gitlab-runner 16.3.0-beta.108.g2b6048b4 (2b6048b4)
2 on blue-6.private.runners-manager.gitlab.com/gitlab.com/gitlab-org
  system ID: s_90e66139ec06
3 feature flags: FF_NETWORK_PER_BUILD:true, FF_USE_FASTZIP:true, FF_U
  ED_URL_MASKING:true, FF_RESOLVE_FULL_TLS_CHAIN:false
4 Resolving secrets
5 Preparing the "docker+machine" executor
6 Using Docker executor with image registry.gitlab.com/gitlab-org/gitlab-build-images/debian-bullseye-slim-ruby-3.1 ...
7 Authenticating with credentials from job payload (GitLab Registry)
8 Pulling docker image registry.gitlab.com/gitlab-org/gitlab-build-images/debian-bullseye-slim-ruby-3.1 ...
9 Using docker image sha256:1451ddfbe810260a6df2f76ec7e8f1d27198f79ee78c4d346e1319b for registry.gitlab.com/gitlab-org/gitlab-build-images/debian-bullseye-slim-ruby-3.1 with digest registry.gitlab.com/gitlab-org/gitlab-build-images/debian-bullseye-slim-ruby-3.1@sha256:0babc6f0f281b4bb5d8592a6c595df4ce7bd6d225577ce02e1b1856 ...
10 Preparing environment
11 Running on runner-nkjtfbei-project-278964-concurrent-0 via runner-nkjtfbei-project-278964-concurrent-0
```

Root cause analysis

The job failed because the git repository could not be fetched. The error message "fatal: remote error: GitLab is currently unable to handle this request due to load" indicates that the GitLab server is experiencing a high load and is unable to process the request. This could be due to a number of factors, such as a large number of users accessing the server at the same time, or a problem with the server itself.

To fix this issue, you could try again later when the load on the GitLab server is lower. If the problem persists, you could contact GitLab support for assistance.

Example Fix

```
# Retry the git fetch command
git fetch

# If the git fetch command still fails, contact GitLab support for assistance.
```



Ultimate

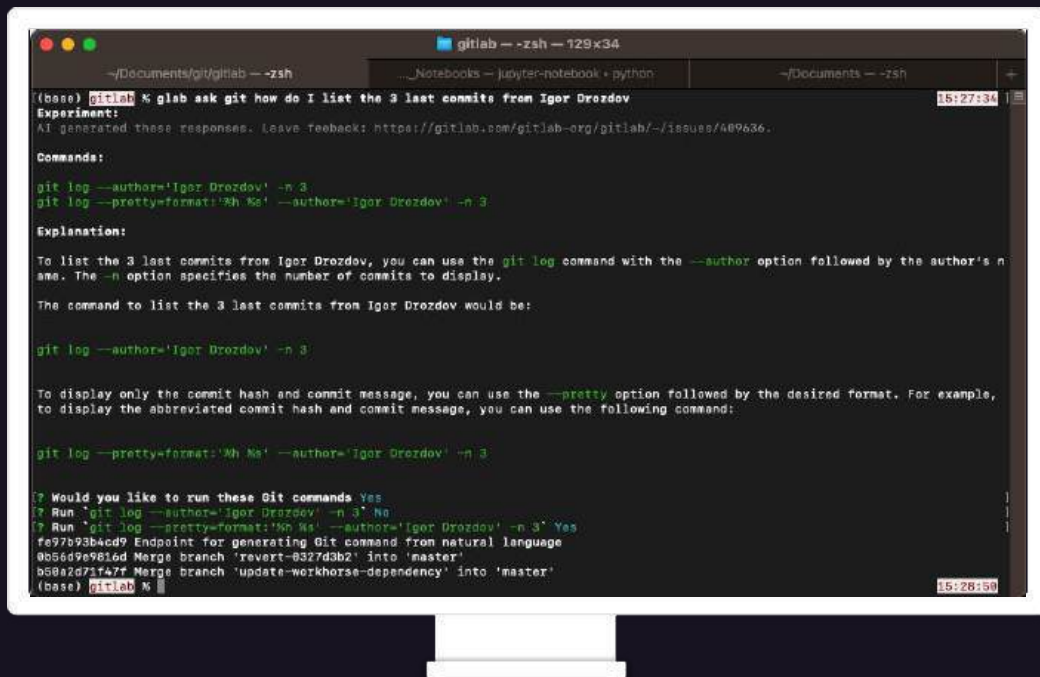
SaaS

Self Managed

GitLab Duo for the CLI

Generate Git commands in GitLab CLI

- ✓ Enables developers to discover or recall git commands where they need them
- ✓ Uplevels developers through continuous learning

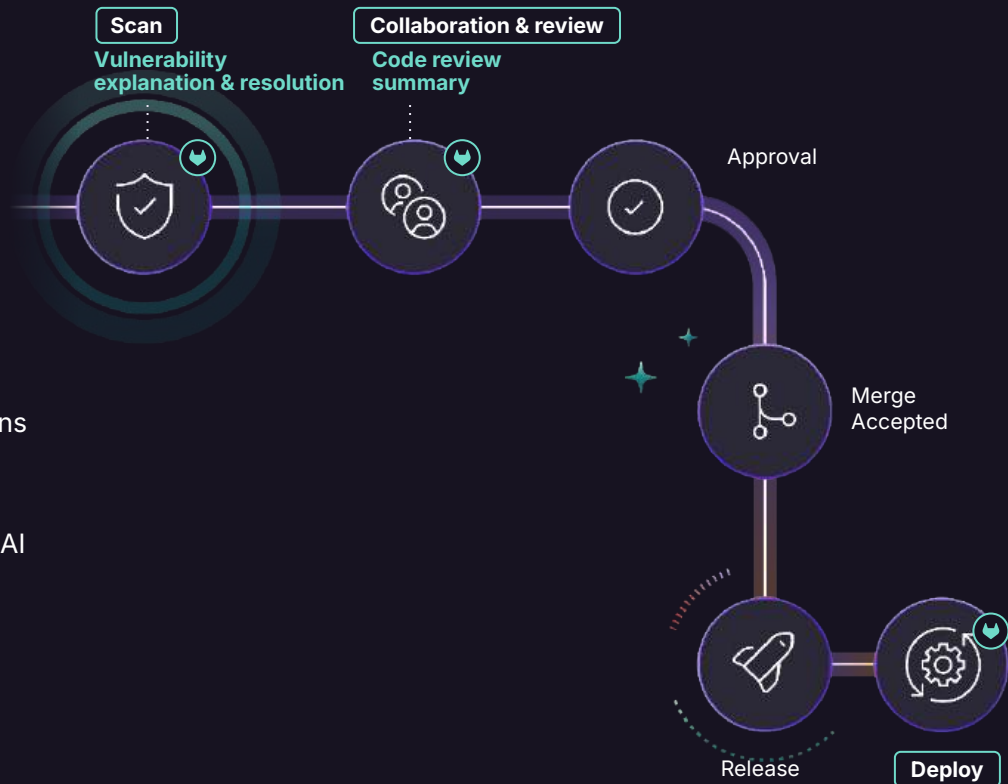




Creating security and governance platform-wide

AI-powered automation for:

- ✓ **Vulnerability explanation**
Explain to developers vulnerabilities and recommendations to prevent vulnerabilities in deployment
- ✓ **Vulnerability resolution**
Automatically open a merge request with recommended AI generated vulnerability remediation
- ✓ **Code review summary**
Ensure better collaboration between developer, security, and operations



Ultimate

SaaS

Self Managed

Vulnerability explanation

Continuous security improvements

- ✔ Helps developers remediate vulnerabilities more efficiently
- ✔ Helps uplevel developers to improve their skills with AI generated security training
- ✔ Enables developers to write more secure code

The screenshot displays a GitLab CI pipeline run for a project named 'govern'. A vulnerability scan tool, Semgrep, has detected a critical issue. The scan results are shown in a table-like format with columns for 'Needs triage', 'Detected', and 'Status'. The detected issue is 'Improper Neutralization of Special Element: Command ('SQL Injection')'. The description explains that the code in 'SQLLiteRoleProvider.cs' is vulnerable to SQL injection due to improper neutralization of user-supplied input. It provides a detailed description of the vulnerability, its severity (Critical), the project path, the tool used (SAST), and the scanner (Semgrep). The location of the issue is identified as 'File: WebGoat/Code/SQLiteRoleProvider.cs:427-429'. The identifiers for the issue are 'SCS0002' and 'security_code_scan.SCS0002-1'. A vulnerability explanation is provided, detailing how an attacker could exploit this vulnerability by crafting a malicious payload. The payload is shown as a code snippet: 'string maliciousRoleName = " OR '1'='1"; cmd.CommandText = "SELECT u.Username FROM " + USER_TB_NAME + " u INNER JOIN r ON u.UserId = r.UserId INNER JOIN role ON r.RoleId = r.RoleId WHERE (r.LoweredRoleName = \$RoleName) AND (r.ApplicationId = " + ApplicationId + "';'.

Feature GA & SM support added in 17.2



Ultimate

SaaS

Self Managed

Vulnerability resolution

Continuous security improvements

- ✓ Helps developers remediate vulnerabilities more efficiently
- ✓ Get a jump start on how to fix a found vulnerability with auto-generated merge requests to mitigate security risks

The screenshot displays a GitLab vulnerability report for a Dockerfile. The report is titled "A user should be specified in the dockerfile, otherwise the i" and is marked as "Needs triage". It was detected 1 year ago in pipeline 8296. The severity is "Critical". The project is "GitLab.org / ModelOps / Demos / Handwritten Digit Recognition" and the scanner used is "kics". The location is "File: Dockerfile:1". The vulnerability is identified as "Missing User Instruction". A suggested resolution is provided: "Resolve with merge request" using GitLab Duo AI to generate a merge request with a suggested solution. The suggested solution is to "FROM nvr.io/nvidia/cuda:12.1.1-base-ubuntu22.04".

GitLab.org / ModelOps / Demos / Handwritten Digit Recognition / Vulnerability Report / 80215951

Needs triage Detected · 1 year ago in pipeline 8296 Status Needs triage

Resolve with merge request

✓ Resolve with merge request
Use GitLab Duo AI to generate a merge request with a suggested solution

Explain vulnerability
Use GitLab Duo AI to provide insights about the vulnerability and suggested solutions

Description
The 'Dockerfile' does not contain any 'USER' instruction

Severity: Critical

Project: GitLab.org / ModelOps / Demos / Handwritten Digit Recognition

Tool: SAST

Scanner: kics

Location
File: Dockerfile:1

```
FROM nvr.io/nvidia/cuda:12.1.1-base-ubuntu22.04
```

Identifiers

- Missing User Instruction

Beta



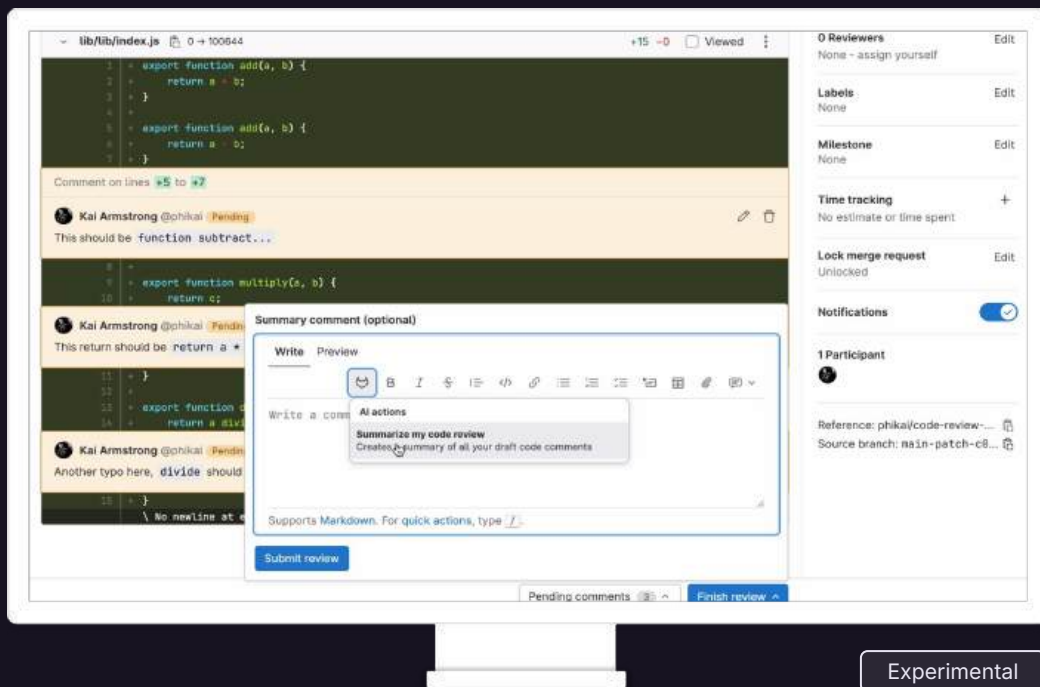
Ultimate

SaaS

Code review summary

Quickly summarize changes you recommend making

- ✔ Enables better handoffs between authors and reviewers
- ✔ Helps reviewers efficiently understand many merge request suggestions



All paid tiers

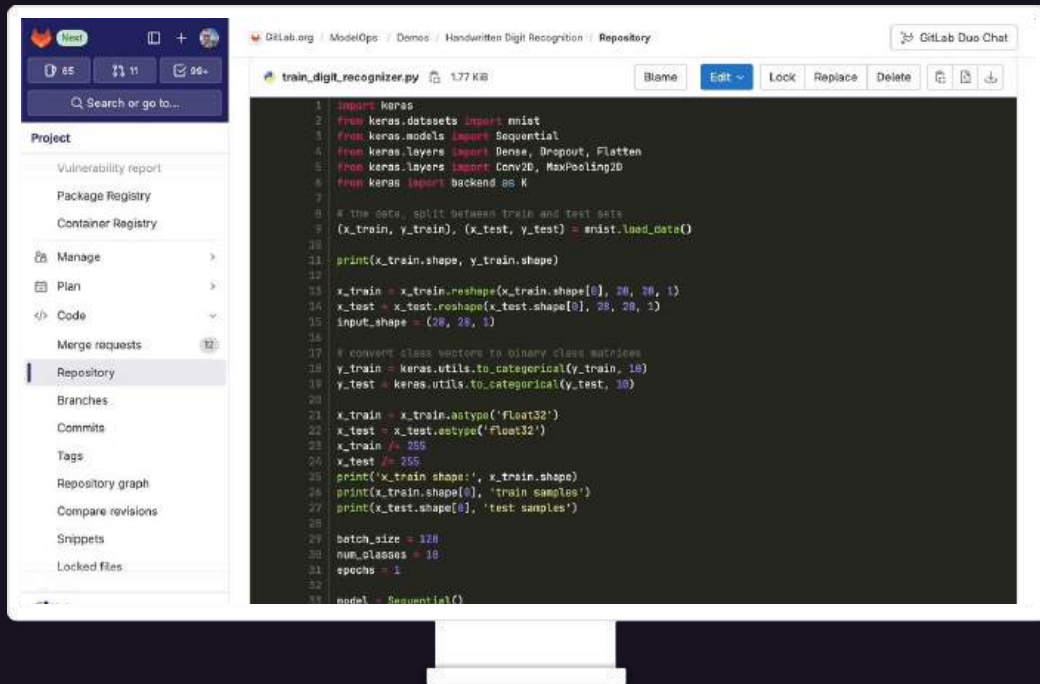
SaaS

Self Managed

Duo chat

Ask questions rather than search for answers:

- ✓ **Code explanation:** Assist in decoding the mysteries of unfamiliar code
- ✓ **Code refactoring:** Supports developers in improving existing code quality
- ✓ **Test Generation:** Automates repetitive tasks for you and helps catch bugs early
- ✓ **Context across Epics, Issues, GitLab Documentation, and soon MRs**



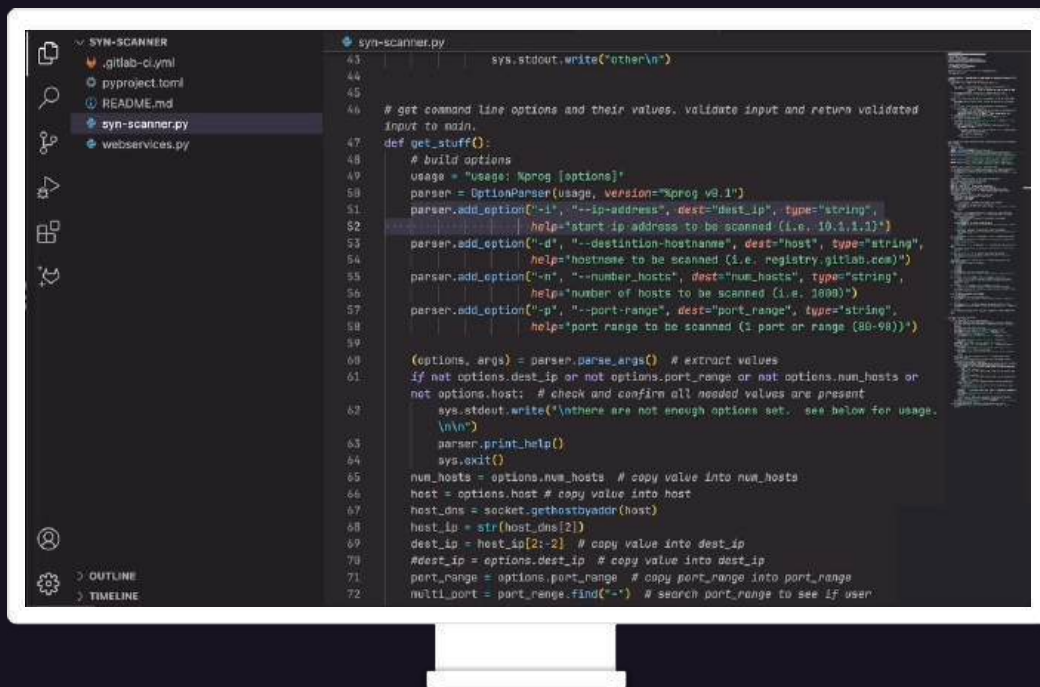
All paid tiers

SaaS

Self Managed

Duo Chat in the IDE

- ✓ Take automated actions on selected code blocks
- ✓ Explain, refactor, and generate tests for code blocks
- ✓ Can answer general and follow up conversations
- ✓ Keeps developers in flow in their editor of choice



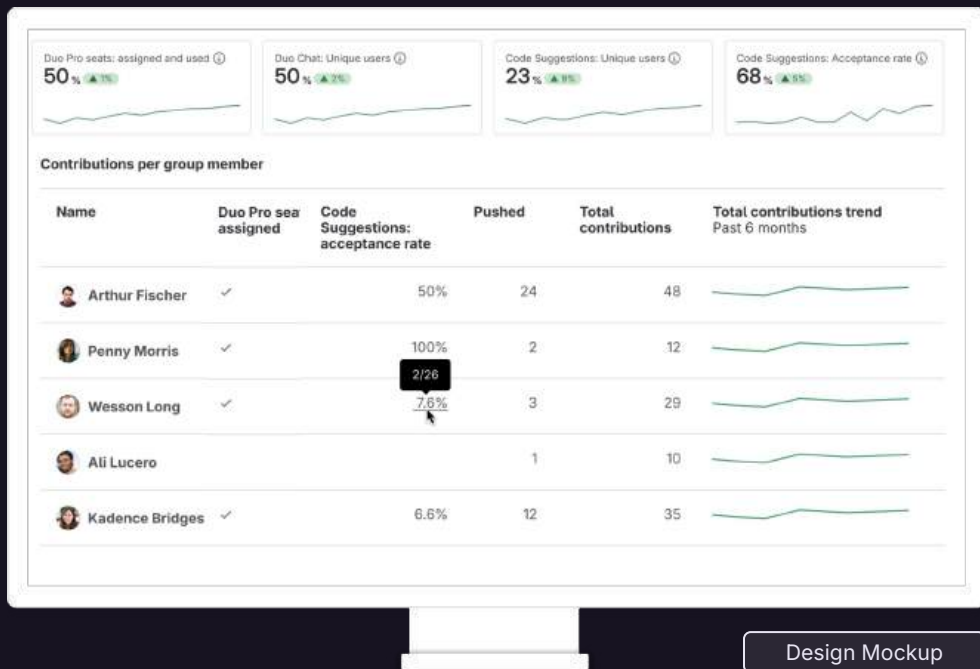
Ultimate

SaaS

Self Managed

AI Impact Dashboard

- ✓ **Correlation observations:** Examine how trends in AI usage within a project or across a group influence other crucial productivity metrics, displayed for the current month and the trailing six months
- ✓ **Comparison view:** Understand the difference in the performance of teams that are and are not using AI, and manage the trade-off between speed, quality, and security exposure



All paid tiers

SaaS

Self Managed

GitLab Duo⁺ access controls

Ensure governance and meet compliance needs

- ✓ Control which group/sub-group/project can use AI features
- ✓ Limit AI on sensitive code or content
- ✓ Assign which users can leverage AI features

GitLab Duo

Configure [AI-powered features](#).

Availability

- On by default
Features are available. However, any group, subgroup, or project can turn them off.
- Off by default
Features are not available. However, any group, subgroup, or project can turn them on.
- Never on
Features are not available and cannot be turned on for any group, subgroup, or project.

GitLab Duo preview features

- Turn on Experiment and Beta features
By enabling these features, you accept the [GitLab Testing Agreement](#).

Save changes



Self Managed

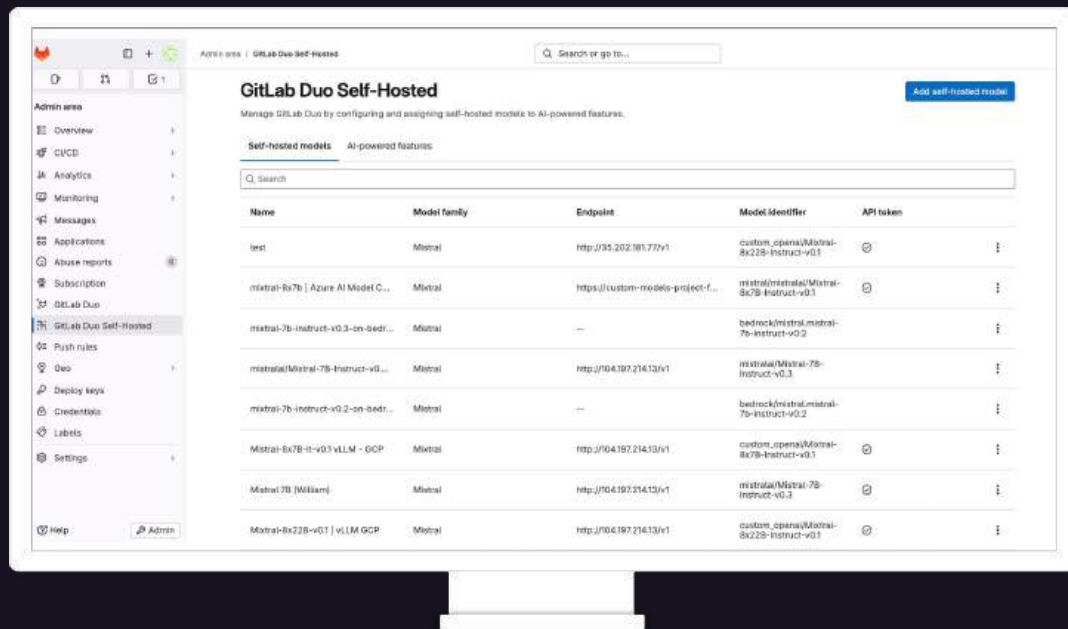
How we differentiate



Self-Hosted

Meet compliance and privacy needs

- ✓ Works completely offline in self managed environment
- ✓ Keeps all AI network requests local
- ✓ Enables completely private AI usage
- ✓ Configure different local models for different features





GitLab Duo+ Regional Gateways Roadmap

Reduced end-user latency to improve user experience



🚩 - Regions Available Today: US East, UK, South Korea, Germany, France

🚩 - Next Planned Regions: US West, Japan

Thank You

Get GitLab
Products Here

Phones

Office : +62-21 5088 6328

Email

admin@gscatalyst.com

Address

Menara Caraka, 7th Floor Jl. Mega Kuningan Barat No.1
Kuningan Timur, Jakarta Selatan, 12950