

# NRU 401 「CodeStream による DevOps を想定したエラー分析対応の基本」

September 13th, 2022

15:00より開始致します。しばらくお待ちください。

音声聞こえづらい場合は退出、再入室をお試しください。

# Safe Harbor

This presentation and the information herein (including any information that may be incorporated by reference) is provided for informational purposes only and should not be construed as an offer, commitment, promise or obligation on behalf of New Relic, Inc. (“New Relic”) to sell securities or deliver any product, material, code, functionality, or other feature. Any information provided hereby is proprietary to New Relic and may not be replicated or disclosed without New Relic’s express written permission.

Such information may contain forward-looking statements within the meaning of federal securities laws. Any statement that is not a historical fact or refers to expectations, projections, future plans, objectives, estimates, goals, or other characterizations of future events is a forward-looking statement. These forward-looking statements can often be identified as such because the context of the statement will include words such as “believes,” “anticipates,” “expects” or words of similar import.

Actual results may differ materially from those expressed in these forward-looking statements, which speak only as of the date hereof, and are subject to change at any time without notice. Existing and prospective investors, customers and other third parties transacting business with New Relic are cautioned not to place undue reliance on this forward-looking information. The achievement or success of the matters covered by such forward-looking statements are based on New Relic’s current assumptions, expectations, and beliefs and are subject to substantial risks, uncertainties, assumptions, and changes in circumstances that may cause the actual results, performance, or achievements to differ materially from those expressed or implied in any forward-looking statement. Further information on factors that could affect such forward-looking statements is included in the filings New Relic makes with the SEC from time to time. Copies of these documents may be obtained by visiting New Relic’s Investor Relations website at [ir.newrelic.com](http://ir.newrelic.com) or the SEC’s website at [www.sec.gov](http://www.sec.gov).

New Relic assumes no obligation and does not intend to update these forward-looking statements, except as required by law. New Relic makes no warranties, expressed or implied, in this presentation or otherwise, with respect to the information provided.

# ウェビナー 各種ご連絡

1. ご質問がある場合は、「Q&A」からご入力ください。



① 画面下  
「Q&A」をクリック！

こちらにご質問をご記入し、  
「送信」をクリックしてください！

②

2. 本日の資料はこの後「チャット」でURLを共有します。アクセスできない場合は、「Q&A」よりお名前とメールアドレスをご連絡ください。

# 自己紹介



## 伊藤 覚宏 (いとう あきひろ)

New Relic 株式会社  
シニアテクニカルサポートエンジニア

- OSS監視ソリューションのテクニカルサポート、クラウド環境の構築・設計、運用設計の経験を活かしお客様をサポートいたします。
  - 専門: 監視、クラウドアーキテクト、AWS、VMware、Zabbix
- インフラエンジニア
- クラウドエンジニア
- サポートエンジニア

# 本日のゴール

- New Relic Code Stream を利用し、顧客体験と連携したアプリ開発、Issue管理を体験する
- Errors in Boxを利用しSREと開発者の円滑なコラボレーションを体験する。
- New Relicにより “One Team”を実現し、顧客満足度の高いサービス開発の実現方法を理解する。

# アジェンダ

時間(目安)	内容
15:00 – 15:45	Code Streamの紹介とハンズオン環境のご案内
15:45 – 16:15	ハンズオン:Code Stream による開発者ロールプレイ
16:15 – 16:30	Errors in Boxの紹介とハンズオン環境のご案内
16:30 – 16:45	ハンズオン: Errors in Box によるSREの役割
16:45 – 17:00	まとめ、アンケートご記入

# CodeStreamとは



# CodeStream連携の3大特徴

## 1. コードベースでの会話可能！！

- ✓ 開発者は **開発ツールから離れることなく** コードと一緒に障害を確認できる

## 1. 「動いているコード」を間違いなく確認可能

- ✓ エラーが発生した **本番環境で動いているものと同じソースコード** を間違いなく確認
- ✓ New Relic - Errors Inbox からの **開発ツールとの連携**

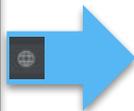
## 3. IDEだけでマルチチャネルのコミュニケーションが可能

- ✓ **開発ツールだけで運用メンバーとコミュニケーション** できる
- ✓ New Relic - 開発ツールの双方向コミュニケーション

# 開発ツールでのNew Relicで観測したエラーの確認

Assigned to Me  
Created by Me (1)  
#1 Add new Socks Created from JetBrains using CodeStream 5h  
Recent (5)  
#1 Add new Socks Created from JetBrains using CodeStream 5h

FEEDBACK REQUESTS  
CODEMARKS (1 total, 1 open)  
OBSERVABILITY (Preview)  
org.springframework.web.util.NestedServletException (6) 4h  
Recent errors in ops  
works.weave.socks.orders.controllers.OrdersController\$PaymentDeclinedException (4) 1d  
org.springframework.web.util.NestedServletException Payment declined: amount exceeded 1d  
org.springframework.web.util.NestedServletException 1d  
works.weave.socks.orders.controllers.OrdersController\$PaymentDeclinedException (3) 1d  
Recent errors in ops  
ISSUES  
Show issues assigned to you from GitHub



Unresolved

works.weave.socks.orders.controllers.OrdersController\$PaymentDeclinedException  
Payment declined: amount exceeded

First seen about 20 hours ago  
Last seen almost 52 years ago  
account: NRK - Demo accountID: 3299505 language: java

Occurrences 6

Stack Trace  
Error message: Payment declined: amount exceeded  
works.weave.socks.orders.controllers.OrdersController.newOrder(OrdersController.java:180)  
sun.reflect.GeneratedMethodAccessor137.invoke(Unknown Source)  
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)  
java.lang.reflect.Method.invoke(Method.java:498)  
org.springframework.web.method.support.InvocableHandlerMethod.doInvoke(InvocableHandlerMethod.java:285)  
org.springframework.web.method.support.InvocableHandlerMethod.invokeForRequest(InvocableHandlerMethod.java:133)  
...  
org.springframework.web.servlet.mvc.annotation.AnnotationMethodHandlerAdapter.handle(AnnotationMethodHandlerAdapter.java:273)  
org.springframework.web.servlet.mvc.annotation.AnnotationMethodHandlerAdapter.handleInternal(AnnotationMethodHandlerAdapter.java:238)  
org.springframework.web.servlet.mvc.annotation.AnnotationMethodHandlerAdapter.handle(AnnotationMethodHandlerAdapter.java:218)

Key Attributes  
transactionName WebTransaction/SpringController... 100.0%  
transactionName /OrdersController/newOrder 100.0%

Feature attributes  
Key Value



CodeStream

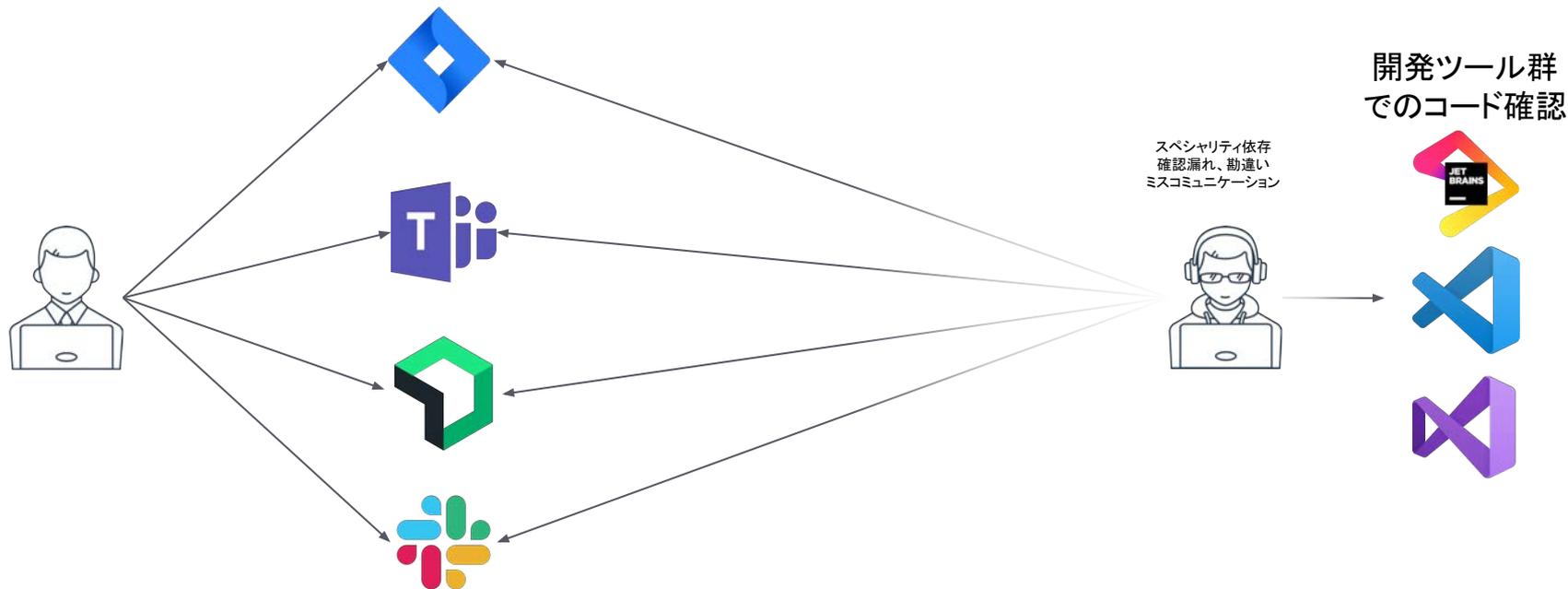
k8s\_catalogue (Ruby) Unresolved

NoMethodError  
undefined method `find' for "brown,formal":String

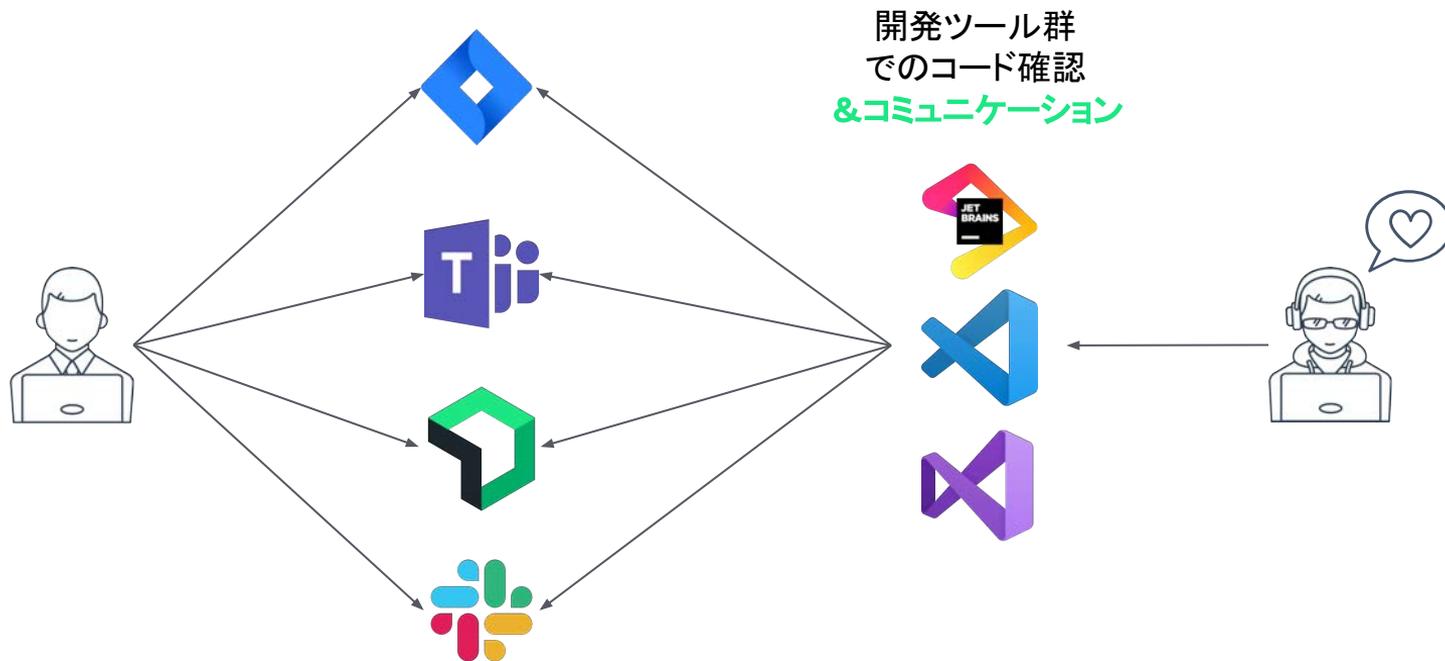
Repo: rcatalogue  
Build: ce939ae

STACK TRACE  
block in call (/home/appuser/webapp/app/services/socks\_selector.rb:13)  
each(/usr/local/bundle/gems/activerecord-6.1.4.1/lib/active\_record/relation/delegation.rb:88)  
each(/usr/local/bundle/gems/activerecord-6.1.4.1/lib/active\_record/relation/delegation.rb:88)  
filter(/home/appuser/webapp/app/services/socks\_selector.rb:12)

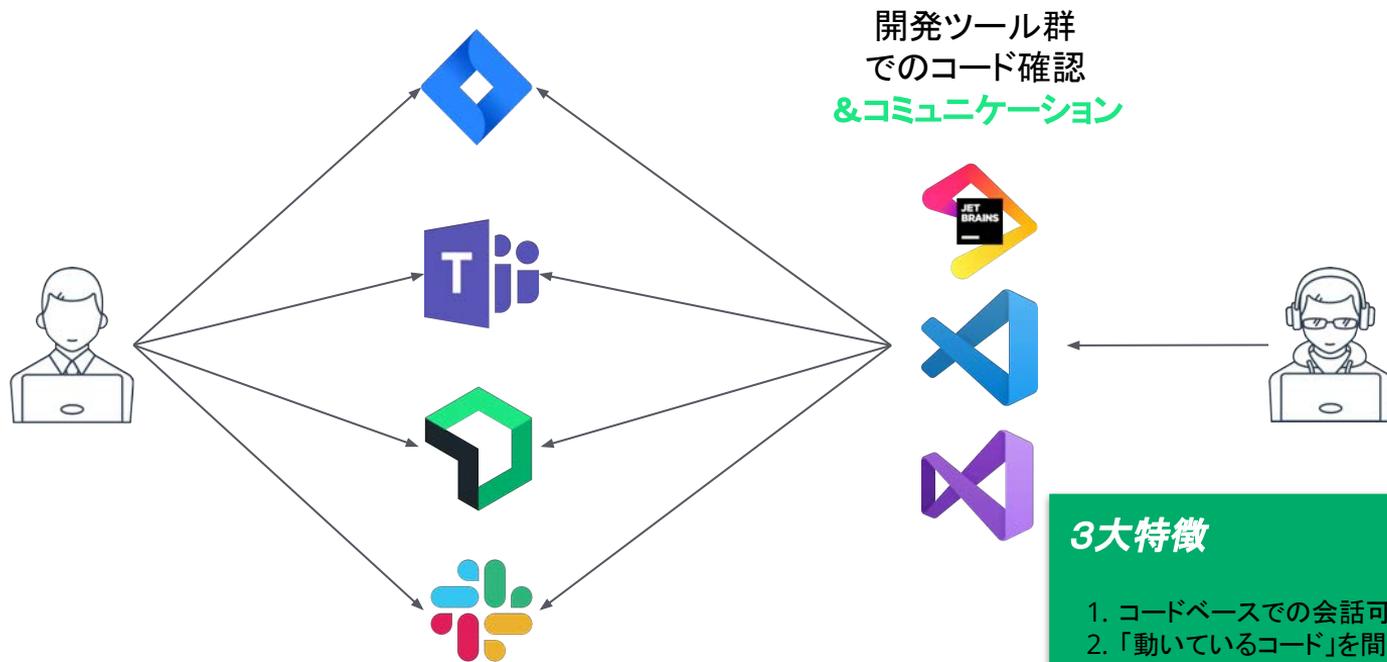
# 人依存の対応がパフォーマンスを妨げている



# 調査から修正までハイパフォーマンスで対応



# 調査から修正までハイパフォーマンスで対応



# New Relic - 開発ツールの双方向コミュニケーション

```
k8s_catalogue (Ruby)
NoMethodError
undefined method `find' for "geek,skin,action,black":String
Repo: rcatalogue
Build: ce939ae
STACK TRACE
block in call(/home/appuser/webapp/app/services/socks_selector.rb:13)
...
filter(/home/appuser/webapp/app/services/socks_selector.rb:12)
call(/home/appuser/webapp/app/services/socks_selector.rb:12)
call(/home/appuser/webapp/app/services/service.rb:5)
size(/home/appuser/webapp/app/controllers/catalogue_controller.rb:17)
...
ACTIVITY
K khara 2 hours ago
エラーが急増しています。カタログ検索APIの稼働率が下がっています。このエラーが関連してると
思われます。調査をお願いしますか？
nrkk-azuma 2 hours ago
このメソッドは配列がくる前提なので、そのエラーが出るのは想定外といえは想定通りなので、呼び
出し元の確認が必要ですね。
smatsukawa 2 hours ago
なるほど。呼び出し元の担当ってどこでしたっけ？
nrkk-azuma 2 hours ago
今日担当の方お休みなので、私みようと思えば見れます。
nrkk-azuma 1 hour ago
エラーバジェットが0なので私担当します。
smatsukawa 1 hour ago
ありがとうございます。確認お願いします。急ぎ目で確認してもらえると助かります。
nrkk-azuma 1 hour ago
リクエストパラメーターで受け取っている値そのまま詰めていますが、これがStringになっているん
ですね。受け取っている文字列見る限りカンマ区切りなのでSplitしようと思います。
app/controllers/catalogue_controller.rb
04. @tags = params[:tags]
```



Occurrences Attributes Activity Discussions (11)

khara@newrelic.com about 2 hours ago  
エラーが急増しています。カタログ検索APIの稼働率が下がっています。このエラーが関連してると思われま  
す。調査をお願いしますか？

Takuya Azuma about 2 hours ago  
このメソッドは配列がくる前提なので、そのエラーが出るのは想定外といえは想定通りなので、呼び出し元の確  
認が必要ですね。

smatsukawa@newrelic.com about 2 hours ago  
なるほど。呼び出し元の担当ってどこでしたっけ？

Takuya Azuma about 2 hours ago  
今日担当の方お休みなので、私みようと思えば見れます。

Takuya Azuma about 1 hour ago  
エラーバジェットが0なので私担当します。

smatsukawa@newrelic.com about 1 hour ago  
ありがとうございます。確認お願いします。急ぎ目で確認してもらえると助かります。

Hide replies

Takuya Azuma about 1 hour ago  
リクエストパラメーターで受け取っている値そのまま詰めていますが、これがStringになっているんですね。受  
け取っている文字列見る限りカンマ区切りなのでSplitしようと思います。

app/controllers/catalogue\_controller.rb

# 開発者と運用者のコミュニケーション例



エラーが大量発生しているので対応をお願いします！

ユーザーに実害出てますか？新機能？

昨日リリースしたので新機能の可能性もありますね

(あー、俺担当じゃ無いやつだ。)バージョンいくつでしたっけ？

2021.10.22.2345のようです。

(バージョンからソースコード逆引きしないと ...) 了解です。

いつ修正バージョン出せそうですか？

(そもそもバグかもわからないし、優先度もわからないし)すぐには答えられそうに無いですね。チームで確認します。



# New Relic x CodeStreamの最大活用例



エラーが大量発生しているので対応をお願いします！

ユーザーに実害出てますか？新機能？

昨日リリースしたので新機能の可能性もありますね

(あー、俺担当じゃ無いやつだ。)バージョンいくつでしたっけ？

2021.10.22.2345のようです。

(バージョンからソースコード逆引きしないと ...) 了解です。

いつ修正バージョン出せそうですか？

(そもそもバグかもわからないし、優先度もわからないし)すぐには答えられそうに無いですね。チームで確認します。

開発やリリースのルールを決めていない場合、対応が難しいことも。コミュニケーション以外で解決できることはたくさんある。  
→CodeStream連携活用！



NR1ダッシュボードやサービスレベル管理機能で状況確認。サービスレベルが決められていると優先順位が明らか。

# New Relic x CodeStreamの最大活用例



商品検索APIのSLOエラーバジェットなくなりそうです。エラーもたくさん出ているので関係ありそうなのですが確認いただけませんか？

それはまずいですね。すぐ動ける人いるか確認して対応します。

(多分担当俺だ...)

(あー、俺担当じゃ無いやつだ。)バージョンいくつでしたっけ？

2021.10.22.2345のようです。

(バージョンからソースコード逆引きしないと ...) 了解です。

いつ修正バージョン出せそうですか？

(そもそもバグかもわからないし、優先度もわからないし)すぐには答えられそうに無いですね。チームで確認します。



NR1ダッシュボードやサービスレベル管理機能で状況確認。サービスレベルが決められていると優先順位が明らか。

開発やリリースのルールを決めていない場合、対応が難しいことも。コミュニケーション以外で解決できることはたくさんある。  
→CodeStream連携活用！

# New Relic x CodeStreamの最大活用例



商品検索APIのSLOエラーバジェットなくなりそうです。エラーもたくさん出ているので関係ありそうなのですが確認いただけませんか？

それはまずいですね。すぐ動ける人いるか確認して対応します。

(多分担当俺だ…)

(ErrorsInbox、Logs、CodeStreamでエラー調査。バージョンを聞かなくてもクリックするだけで目的のソースがわかるからすごい！)

これ新機能追加したときのデグレだ…

いつ修正バージョン出せそうですか？

エラーバジェット見る限りすぐ修正した方がいいと思うので暫定版すぐ出せるよう調整します。

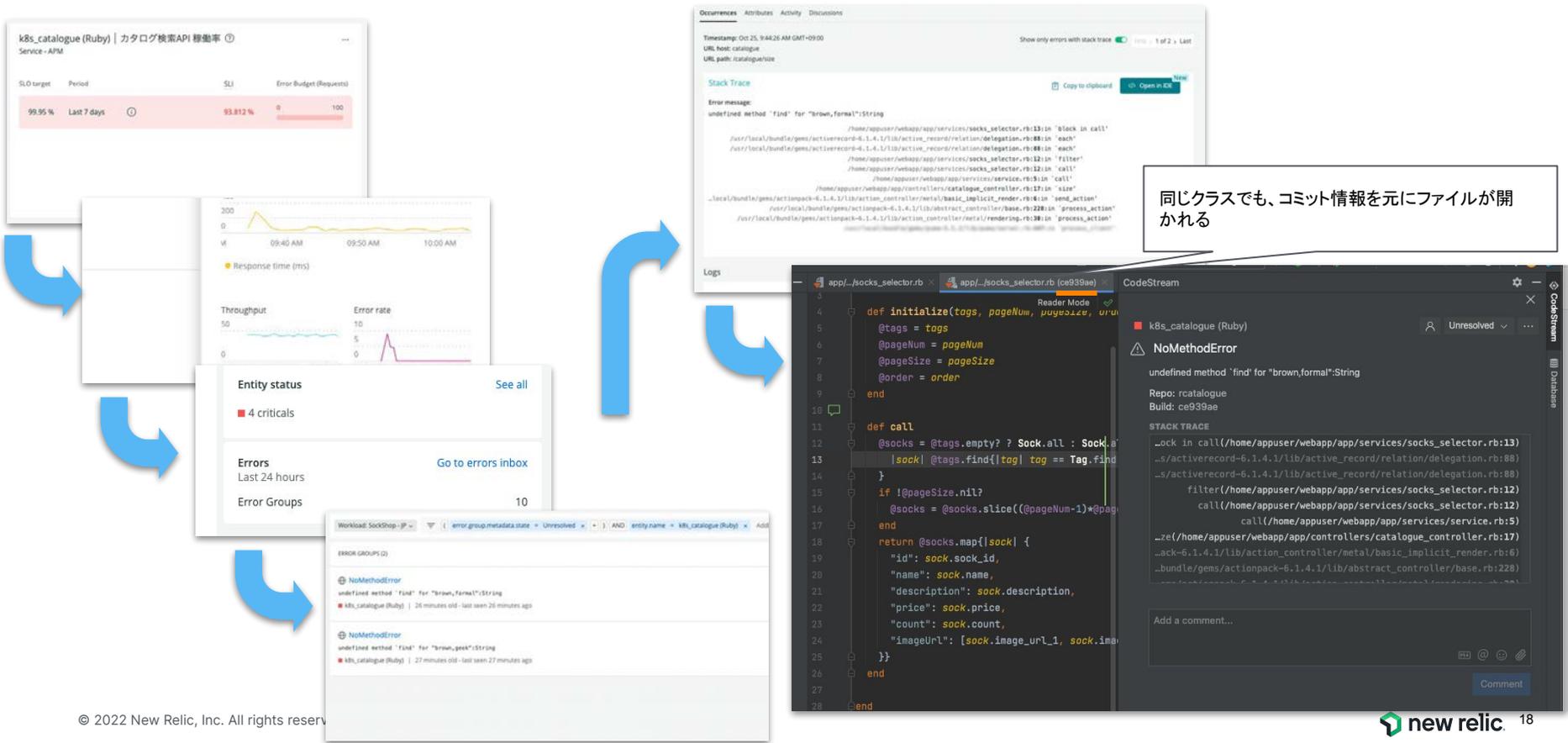
わかりました。よろしくお願いします。



NR1ダッシュボードやサービスレベル管理機能で状況確認。サービスレベルが決められていると優先順位が明らか。

開発やリリースのルールを決めていない場合、対応が難しいことも。コミュニケーション以外で解決できることはたくさんある。  
→CodeStream連携活用！

# 画面利用シーケンス例



# CodeStream連携利用の想定ベストケース

## 1. 開発チームと保守チームがバラバラのケース

自分たちで開発していないので、ソースコードの構成を全部把握していない。

ソースコードをみなくても **1クリック**で目的のコードに到達。

## 2. 開発/保守チームと運用チームのコミュニケーションがメールやチャットなど

デプロイされているバージョンの正確な把握ができていない可能性も。

コミット情報に結びつけて確認できるので、ミスコミュニケーションを大幅削減

## 3. 複数アプリの障害・非機能要件確認をコードベースで行う SRE

複数のリポジトリを一度に管理可能。関連するコードに1クリックアクセスできるので、

バージョン管理ツールや Grepよりも素早くコードに到達し、生産性が向上

CODESTREAM

michael ▾



OBSERVABILITY (Preview)



Errors assigned to me

△ com.newrelic.distributedtracingservice.api.model.APIException

Recent errors in distributed\_tracing\_service distributed\_tracing\_se...

△ com.newrelic.distributedtracingservice.api.model.APIExc... 47m

△ org.apache.http.client.ClientProtocolException (1) 6h

△ java.util.concurrent.CompletionException (1) 6h

△ java.util.concurrent.CompletionException (1) 11h

△ org.apache.http.client.ClientProtocolException (1) 1d

Show All Commands

Go to File

Find in Files

Start Debugging

Toggle Terminal

&gt; FEEDBACK REQUESTS

&gt; CODEMARKS

&gt; ISSUES



# ここがアツいよ！ CodeStream

ここがアツい！① **IDEですべてが完結！**

# ここがアツいよ！ CodeStream

ここがアツい！① **IDEですべてが完結！**

ここがアツい！② **流れるようにコードで会話！**

# ここがアツいよ！ CodeStream

ここがアツい！① **IDEですべてが完結！**

ここがアツい！② **流れるようにコードで会話！**

ここがアツい！③ **アプリの問題コードがすぐわかる！**

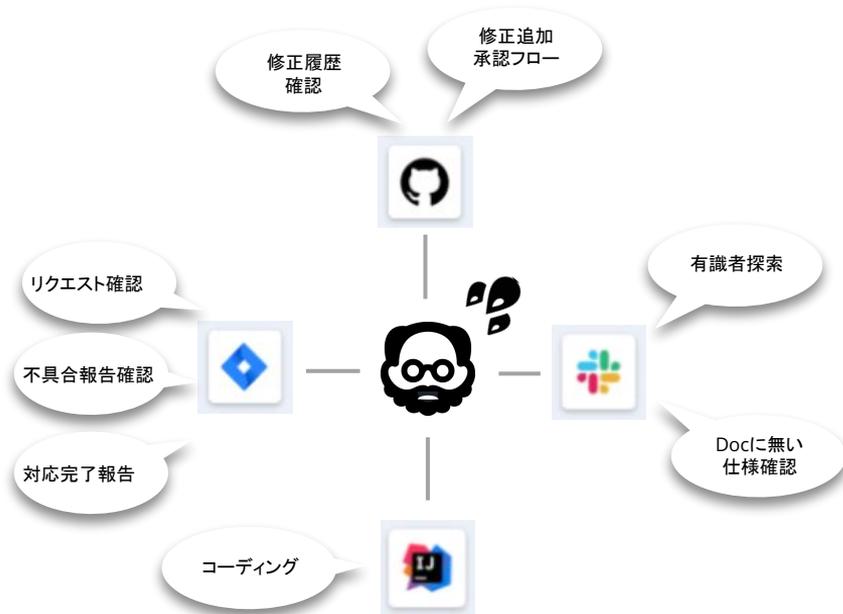
# ここがアツいよ！ CodeStream

ここがアツい！① **IDEですべてが完結！**

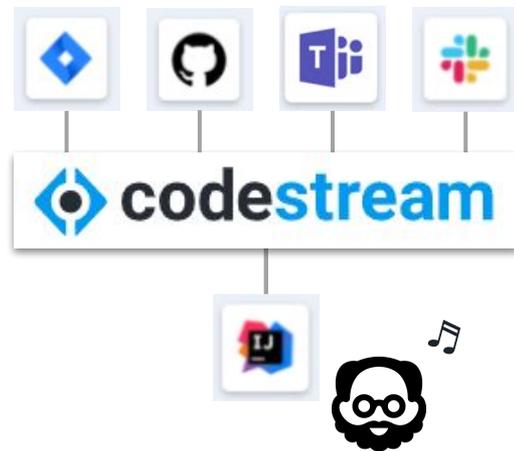
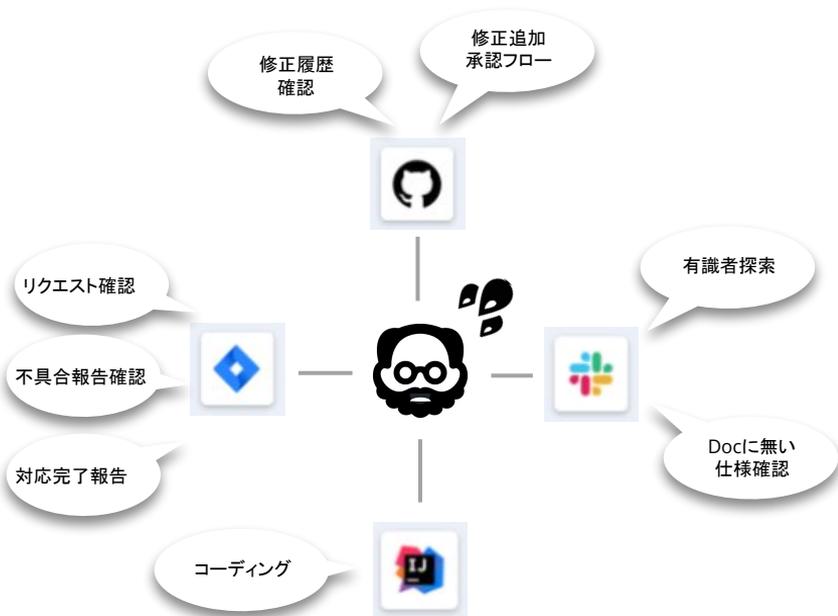
ここがアツい！② **流れるようにコードで会話！**

ここがアツい！③ **アプリの問題コードがすぐわかる！**

# さまざまな便利ツールを駆使した開発...



# 開発環境に全てを統合！



# 多様な開発環境にフィット！

## IDE



VS Code



VS Code  
Insiders



Visual  
Studio



Android  
Studio



IntelliJ  
IDEA



PyCharm



WebStorm



PhpStorm



RubyMine



Rider



CLion



GoLand



DataGrip



AppCode

## Issue管理



GitHub



GitLab



JIRA



Azure  
DevOps



Bitbucket



asana

Asana



Linear



Trello



YouTrack

## ソース管理



GitHub



GitLab



Bitbucket

## メッセージング



Slack



Teams

## タスクの管理からコードのマージまで一連の作業がIDE上で完結

- GitHubやJIRAなどと連携してIDEからIssueやTaskの作成と管理が可能
- 修正ブランチも自動作成し、仕掛かり中のタスクとブランチを自動連動
- PR作成、レビュー、マージもIDE内で完結
- コンテキストを維持しているのでIssueやブランチとも自動で関連付け

- GitHubやJIRAなどと連携してIDEからIssueやTaskの作成と管理が可能
- 修正ブランチも自動作成し、仕掛かり中のタスクとブランチを自動連動
- PR作成、レビュー、マージもIDE内で完結
- コンテキストを維持しているのでIssueやブランチとも自動で関連付け

エラーチェックルールの外だし **Issue作成**

エラーチェックがハードコードされているので変更に弱い。ポリシー外出しにしておく。

[complex-app-settings] app\_settings/app/controllers/settings\_controller.rb (Lines 13-14)

```
01.
02.     if @setting.key.start_with?('error')
03.         error_loop1("Intended Error: " + @setting.key)
```

<sup>Created from VS Code using [CodeStream](#)</sup>

Set up a branch in `complex-app-setitngs` ▾

master ▾  
① 2 commits behind origin **Pull**  
feature/エラーチェックルールの外だし ▾

Update my **Open a Pull Request** **PR作成**

Choose two branches to start a new pull request.

エラーチェックルールの外だし #19 **PRレビュー、マージ**

**Open**

ktst79 wants to merge 1 commits into complex-app-setitngs:master from feature/エラーチェックルールの外だし 4 minutes ago

もともとハードコードされていた入力チェックのルールを外出しにして、変更に対応できるようにする。

**This PR Addresses:**

エラーチェックルールの外だし

Created from VS Code using [CodeStream](#)

Put error check outside the code 6e152c0

Add more commits by pushing to the feature/エラーチェックルールの外だし branch on ktst79/complex-app-setitngs.

Some checks haven't completed yet [Show all checks](#)  
1 in progress check

**✓ This branch has no conflicts with the base branch**  
Merging can be performed automatically.

**Squash and merge** ▾

repo: complex-app-setitngs ▾ base: master ▾

compare: feature/エラーチェックルールの外だし ▾

エラーチェックルールの外だし

もともとハードコードされていた入力チェックのルールを外出しにして、変更に対応できるようにする。

Set upstream to origin/feature/エラーチェックルールの外だし

This PR addresses: [エラーチェックルールの外だし](#)

Cancel **Create Pull Request**

# ここがアツいよ！ CodeStream

ここがアツい！① IDEですべてが完結！

ここがアツい！② 流れるようにコードで会話！

ここがアツい！③ アプリの問題コードがすぐわかる！

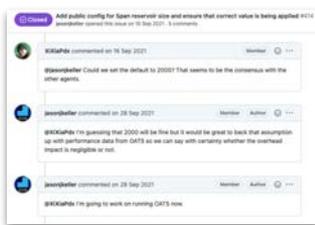


# IDE上でコードについて直接会話！

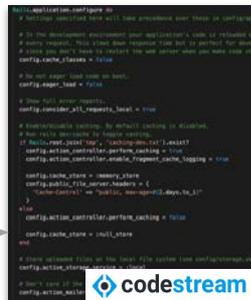
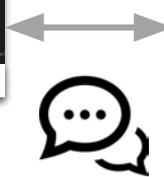
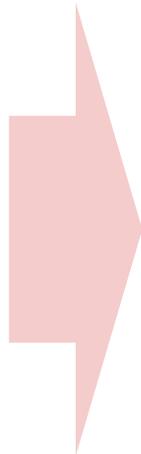
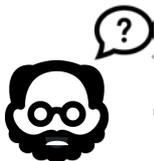
チャットで聞いて回る



Issue/PRを漁る



Web会議でレビュー



# コードに関する やりとりをIDE上で完結

- コード上で直接ディスカッションを開始して会話が可能
- 修正者を探さなくても自動でメンション（もちろん他の人も可）
- 会話に関するコードを探すことなく、コメントから自動表示
- PRやIssueがクローズされてもコードと関連づいて残るので過去の経緯の把握が可能
- SlackやTeamsなどのコミュニケーションツールにも自動で同期

The screenshot displays a code editor on the left and a chat window on the right. The code editor shows a Ruby class `SettingsController` with methods `index`, `new`, `create`, `edit`, `show`, and `update`. A comment in the `create` method reads: `// ktst79: @jWick ここにあるエラーハンドリングってどういう意図ですか?`. The chat window shows a conversation between `ktst79` and `jWick`. `ktst79` asks about the error handling logic. `jWick` explains it's an input field validation check. `ktst79` then asks why it's hard-coded, and `jWick` replies that it's a consideration for future changes.

```
app_settings > app > controllers > settings_controller.rb
1 class SettingsController < ApplicationController
2   def index
3     @settings = Setting.all
4   end
5
6   def new
7     @setting = Setting.new
8   end
9
10  def create
11    @setting = Setting.new(setting_params)
12
13    // ktst79: @jWick ここにあるエラーハンドリングってどういう意図ですか?
14    if @setting.key.start_with?('error')
15      error_loop1("Intended Error: " + @setting.key)
16    elsif @setting.key == 'slow'
17      #Some comment...
18      100.times do
19        dummy = Setting.all
20      end
21    end
22
23    if @setting.save
24      redirect_to settings_path
25    else
26      render 'new'
27    end
28  end
29
30  def edit
31    @setting = Setting.find(params[:id])
32  end
33
34  def show
35    @setting = Setting.find(params[:id])
36  end
37
38  def update
39    @setting = Setting.find(params[:id])
40
41    if @setting.key == 'error'
42      error_loop1("Intended Error")
43    end
44  end
45 end
```

CHAT WINDOW: CODEMARKS (1) complex-app-settings

**ktst79** 25 minutes ago

@jWick ここにあるエラーハンドリングってどういう意図ですか?

SHARED TO #nr\_cs\_ktst

complex-app-setitngs  
app\_settings/app/controllers/settings\_controller.rb  
master 74c6655

```
13.   if @setting.key.start_with?('error')
14.     error_loop1("Intended Error: " + @setting.key)
```

ACTIVITY

**jWick** 24 minutes ago @ktst79 入力フィールドの不正チェックです。

**ktst79** 23 minutes ago @jWick ハードコードになっちゃっていて仕様変更に対応できないのですが、外だししていない理由とかありますか？

**jWick** 22 minutes ago @ktst79 いや、待たないです。考慮漏れなので直してもらえると助かります。

**ktst79** 22 minutes ago @jWick ラジャー

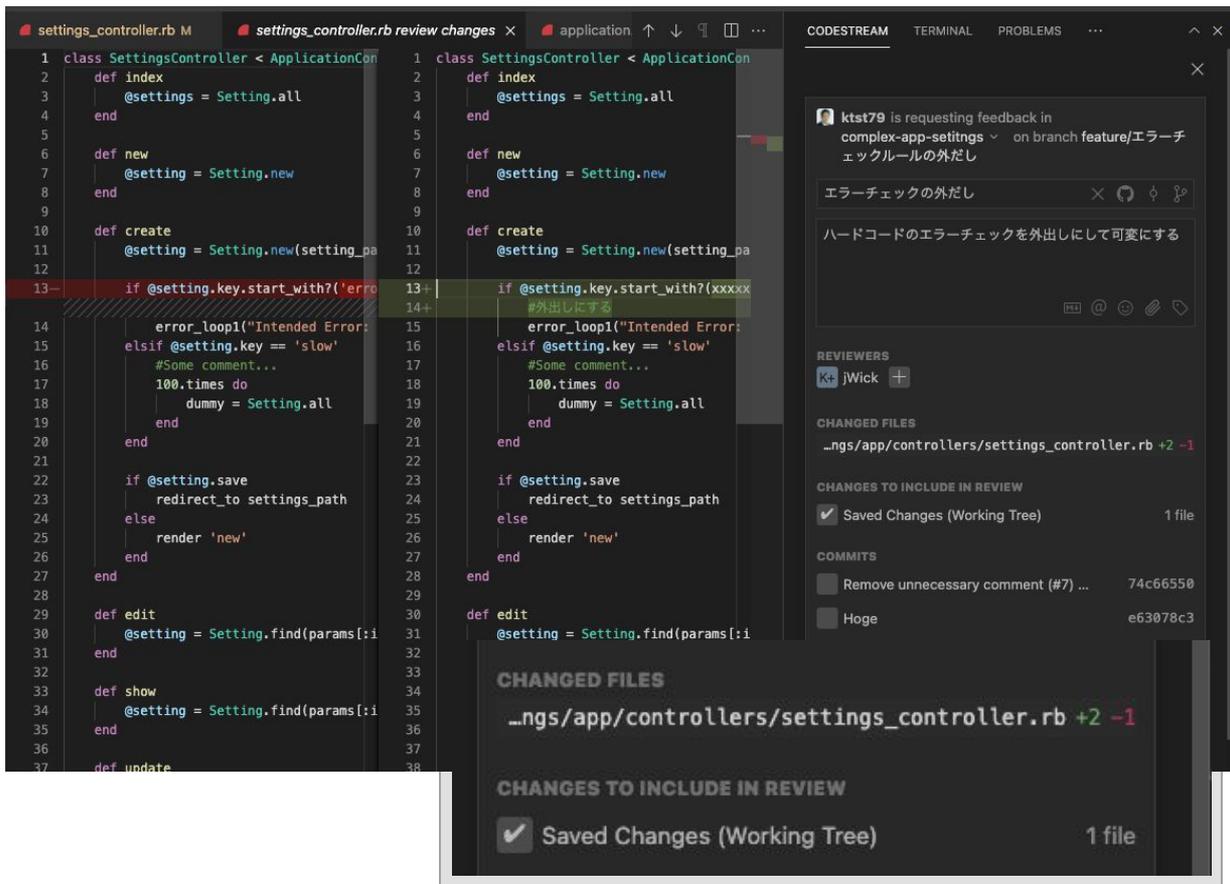
ADD REPLY

Reply...

Resolve & Archive Discussion Comment

# 未コミットの修正の レビューによる レビューの軽量化

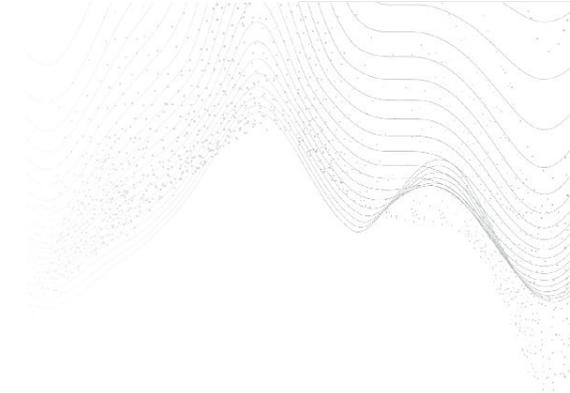
- コミット前のローカルの修正のレビューが可能(もちろんコミット済みも)
- レビュープロセスの軽量化と前倒しによる、後フェーズでの手戻り抑止とレビューアの負荷分散



# “エンジニア”にはこんな世界が待ってる

- 開発チームは運用チームとのコミュニケーションコストを削減し、**人い依存なく、短時間で正確なバグ修正が可能になる**
  - 高度な技術知識とコミュニケーションスキル
  - ソースコード構成の把握
  - アプリケーションの理解
- 開発エンジニアは、**本来の業務であるプロダクトの開発**に時間を使えるようになる
- コードレベルの問題解決速度の向上により、運用チームと開発チームの**負担を最小限にリリースサイクルを早める**事ができる

# 事前作業

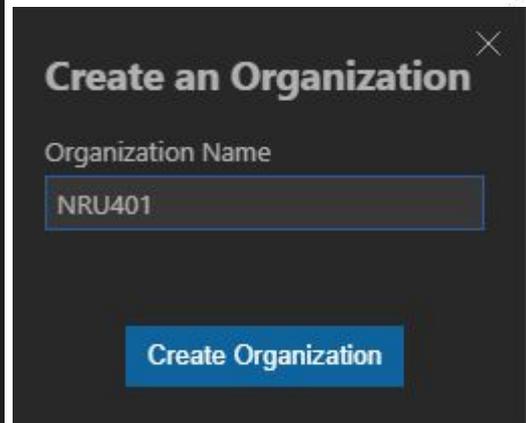
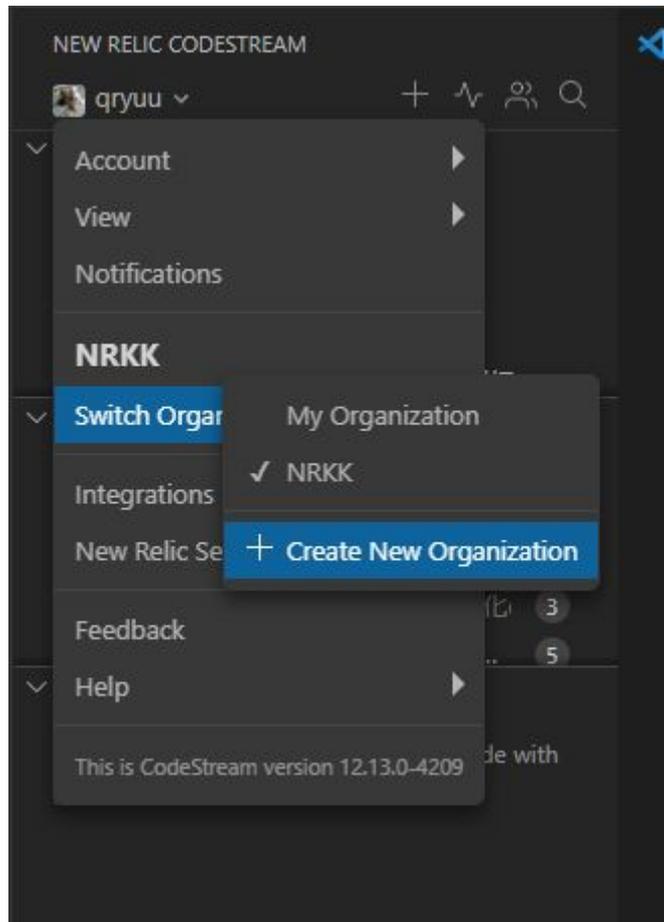


ここでは開発チームで最初に 1人だけ実施する作業を紹介しています。

ハンズオン環境では既に実施済みです。

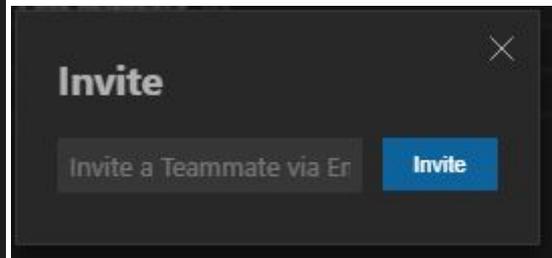
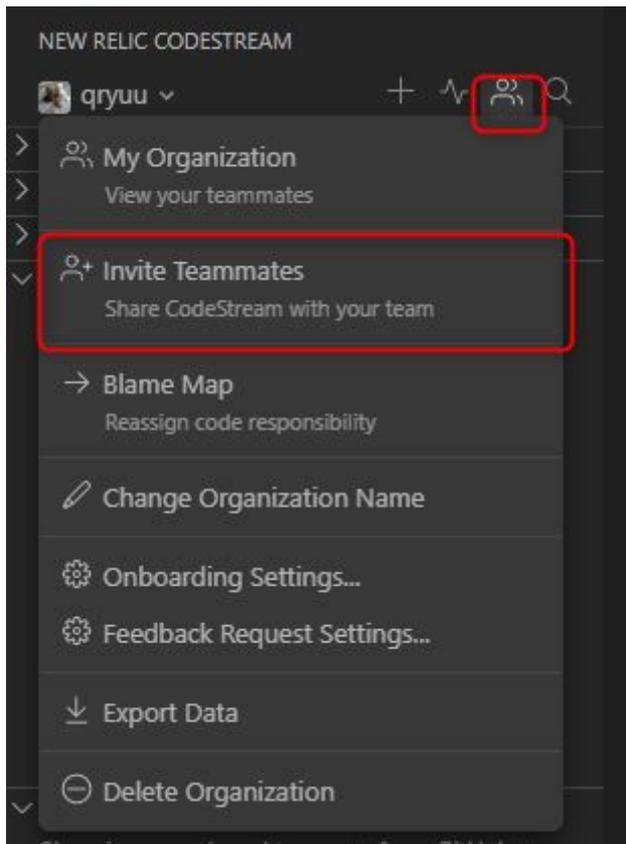
# 参考 Organizationの 作成

はじめにCode Stream  
Organizationの作成を行います。  
この操作は開発リーダーなどの  
1人の人が行う作業のため、  
参加者の方は実施不要です。  
既に作成済みです。



# 参考 Organization へのUser追加

Code Stream Organizationに  
チームメンバーを登録します。  
Code Stream はメールアドレス  
でUserを管理します  
今回は参加者の皆さんメール  
アドレスを追加済みです。



# Code Stream

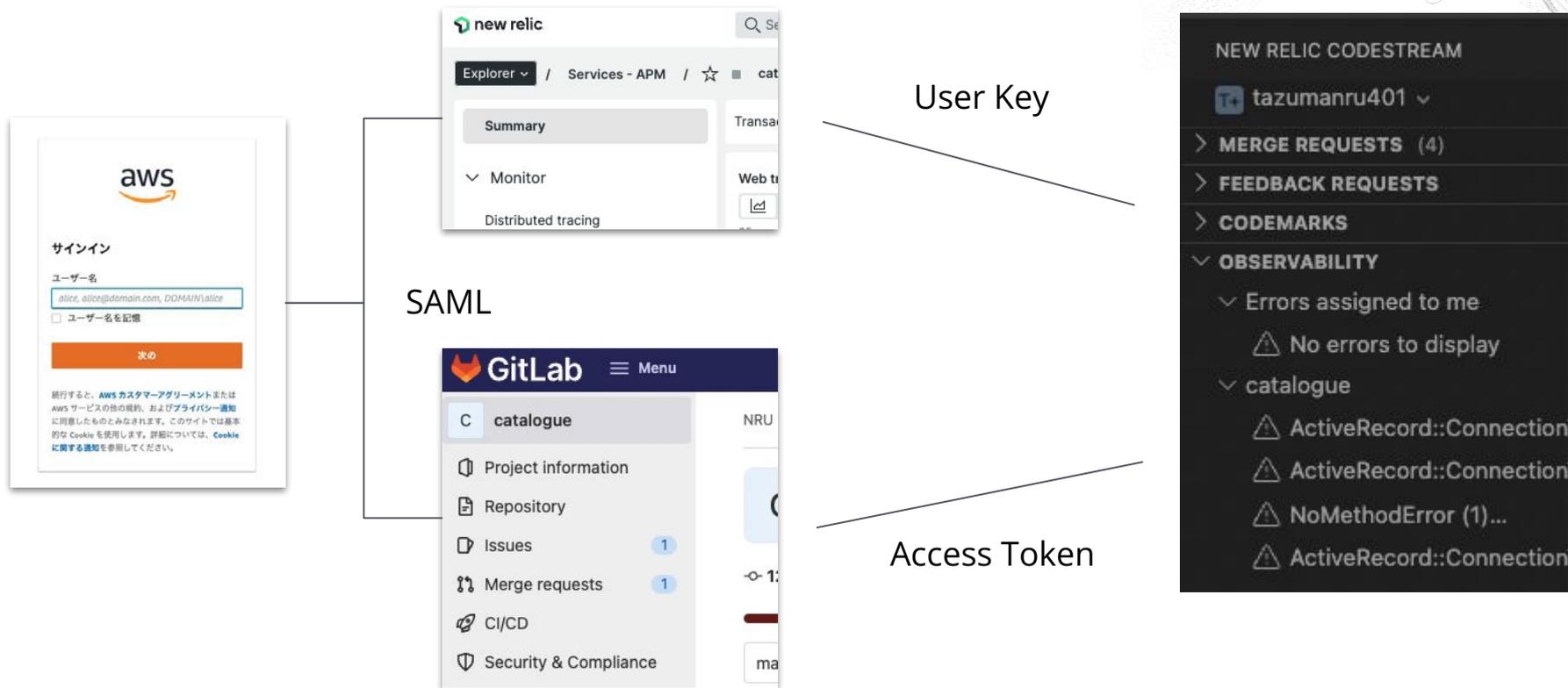
## ハンズオン

# 前提条件

本ハンズオンではIDEとして、Visual Studio Codeを前提としています。  
異なるIDEをご利用の方は適宜読み替えてください。

CodeStreamの機能、使い方についてのハンズオンとなります、個別の Codeの記法、書き方については扱いません

# アカウント構成図

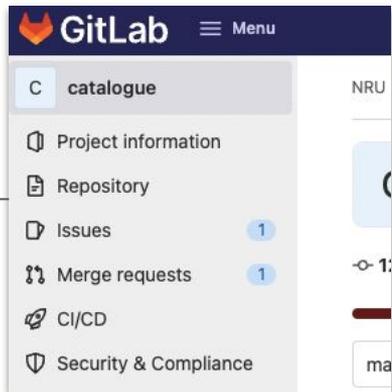
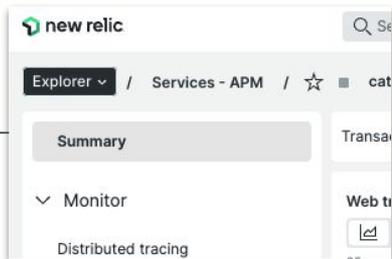


# アカウント構成図

運営側より払い出し



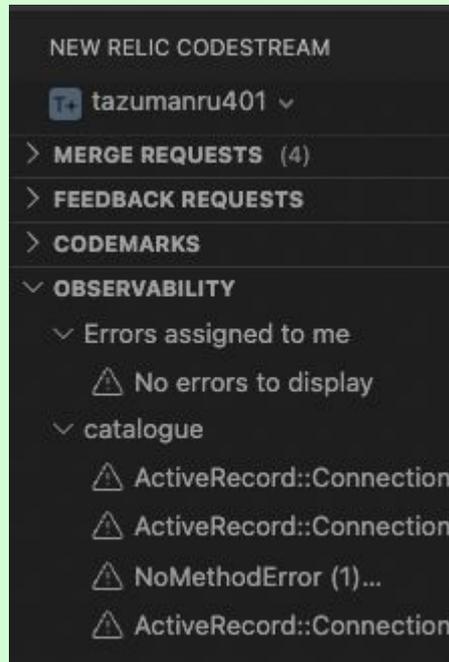
SAML



Invitationからログイン

User Key

Access Token



ハンズオンで作業

# AWS SSOへのログイン

## 1. ログイン画面へアクセス

<https://d-9567758112.awsapps.com/start>

ユーザー名: nru401user0\*\*

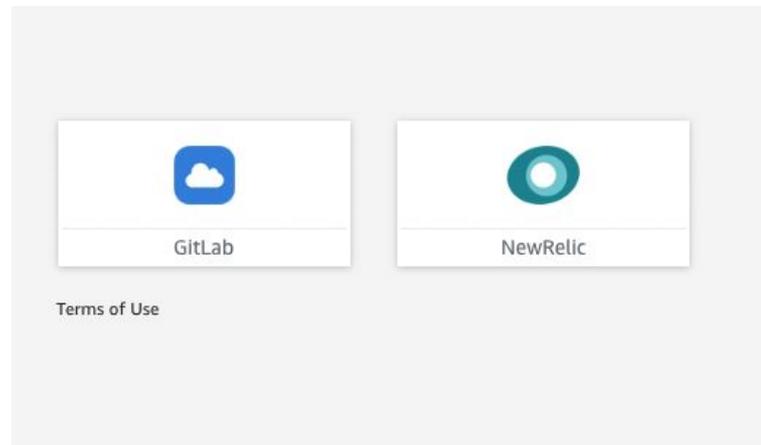
パスワード: dfWet24e6sX!63d



## 2. アプリ登録状況の確認

GitLabおよびNew Relicのアプリが登録されていることを確認してください。

どちらもアクセスしてみましょう。



# GitLabログイン

GitLabにアクセスしたら、Sign in with「AWS SSO(Japan NRU)」ボタンをクリックしてログインします。

GitLab

**A complete DevOps platform**

GitLab is a single application for the entire software development lifecycle. From project planning and source code management to CI/CD, monitoring, and security. This is a self-managed instance of GitLab.

Username or email

Password

Remember me [Forgot your password?](#)

Sign in

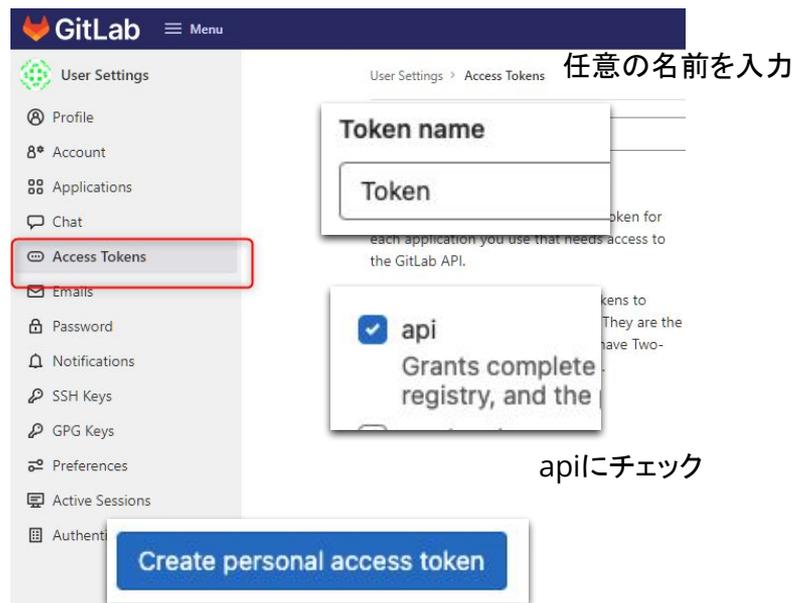
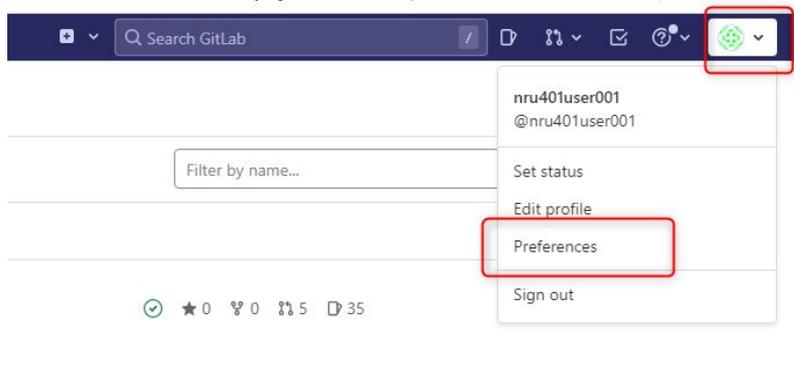
Sign in with

Remember me

AWS SSO (Japan NRU)

# 各キーの準備 1/2 (GitLab)

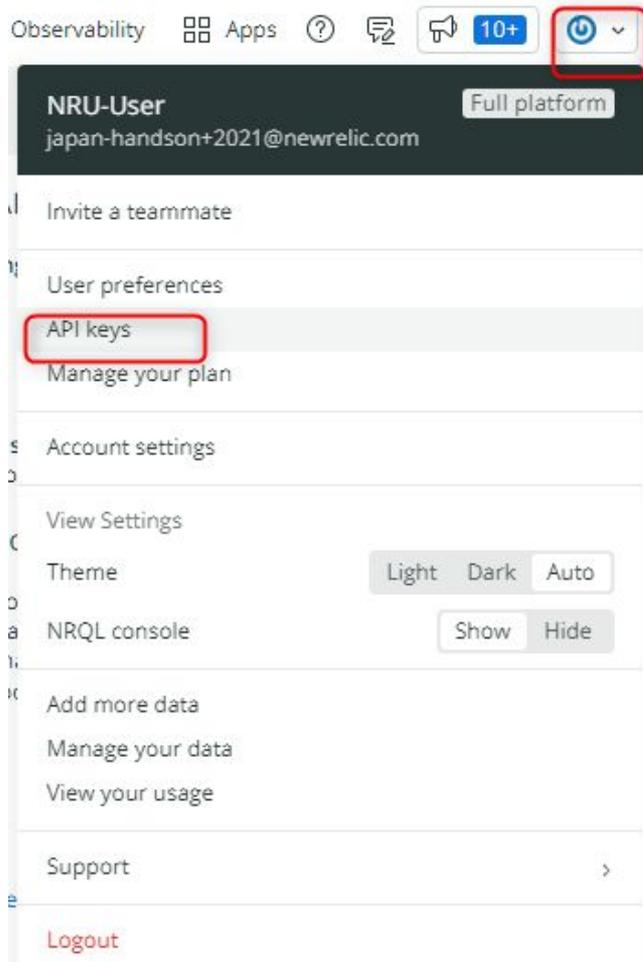
CodeStreamのGitLab連携、Gitへのコミット用にGitLabのアクセストークンを発行します。  
作成して保管をお願いします (\*ハンズオン後こちらで破棄します)



# 各キーの準備 2/2 (New Relic)

Code Stream とNew Relic Accountを接続します。  
Code Stream にNew Relicの UserKeyを登録します。

API KeyからKeyを発行してください。



Observability Apps ? 10+ [Power icon]

NRU-User Full platform  
japan-handson+2021@newrelic.com

- Invite a teammate
- User preferences
- API keys**
- Manage your plan
- Account settings
- View Settings
- Theme Light Dark Auto
- NRQL console Show Hide
- Add more data
- Manage your data
- View your usage
- Support >
- Logout

## Create an API key

Ingest keys are for getting data into New Relic:

- License keys for agent configuration and metric, event, log and trace APIs
- Browser keys for browser applications
- Mobile keys for mobile applications

User keys are for querying data and managing configurations (Alerts, Synthetics, dashboards, etc.)

To learn more about API keys, [see our docs](#)

### Account

Account: 2511671 - NewRelicUniversity-Japan

### Key type

User

### Name

NRU401-20220324-YourName

### Notes

NRU401-20220324-YourName

Cancel

Create a key

# CodeStream アカウントへのログイン

## 1, Invitationメールの確認

「<メールアドレス> invited you to collaborate」というメールが届いている事を確認します。

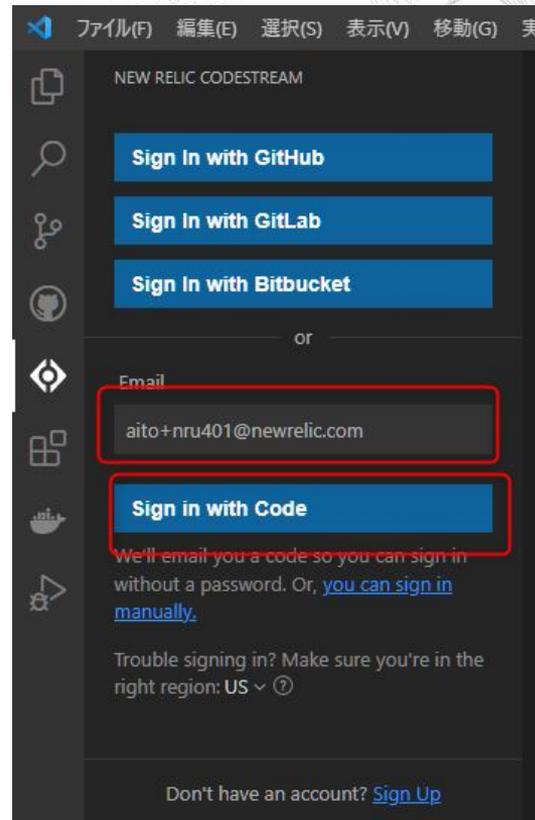
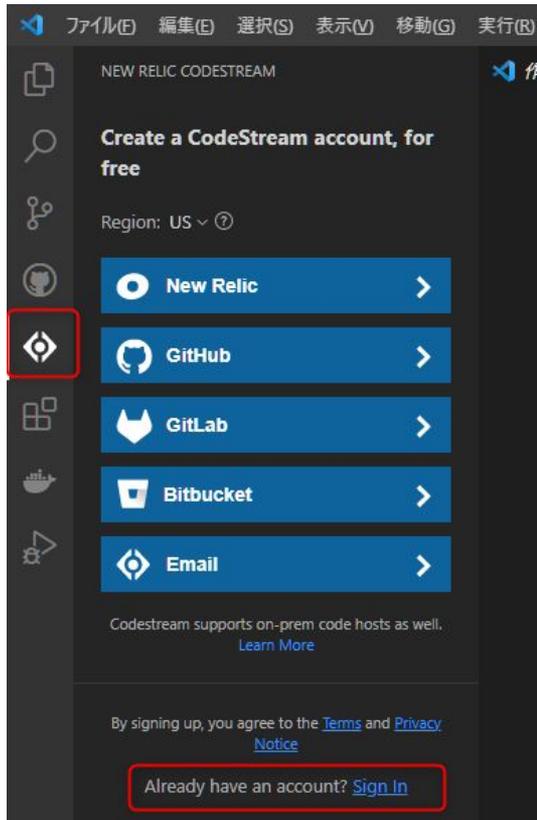


# CodeStream アカウントへのログイン

## 2, アドレス入力

IDEでCodeStream プラグインを開きます。

Sign up 画面が表示される場合は「Sign in」をクリックして、メールアドレスを入力し「Sign in with Code」をクリックします。



# CodeStream アカウントへのログイン

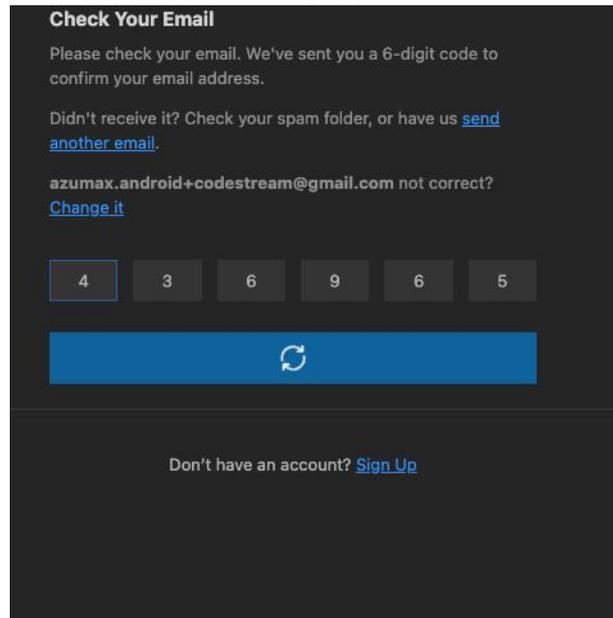
## 3, 6-digitコードの確認

メールアドレスに対して6桁のワンタイムパスワードが届きます。



## 4, コードでSign in

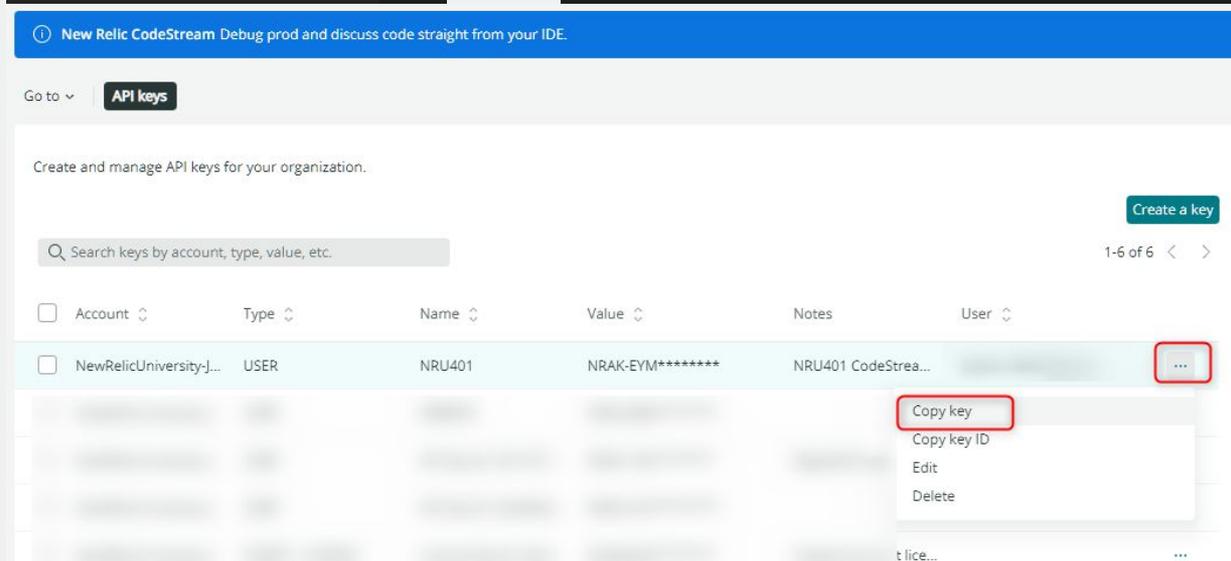
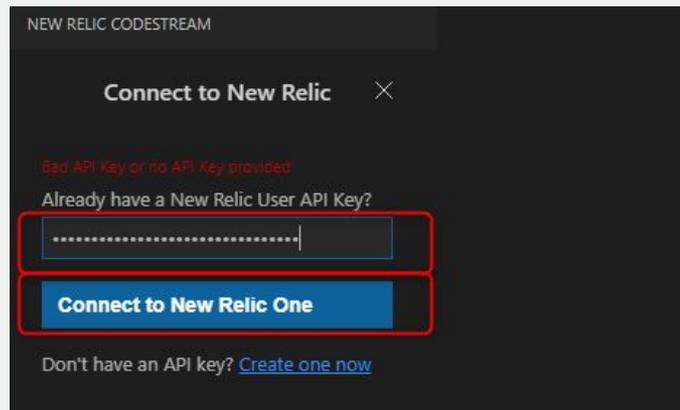
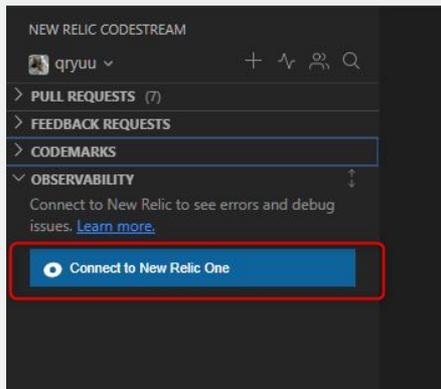
CodeStream プラグインに受信したワンタイムパスワードを入力してログインします。



# New Relicとの 接続

Code Stream とNew Relic  
Accountを接続します。  
Code Stream にNew Relicの  
AccessKeyを登録します。

これにはCoreUser以上の  
Licenseが必要です。



# CodeStreamとGitLabの連携

1, GitLab Self-Managedを選択



2, GitLabのURL、Access Tokenを貼り付け

### Connect to GitLab Self-Managed

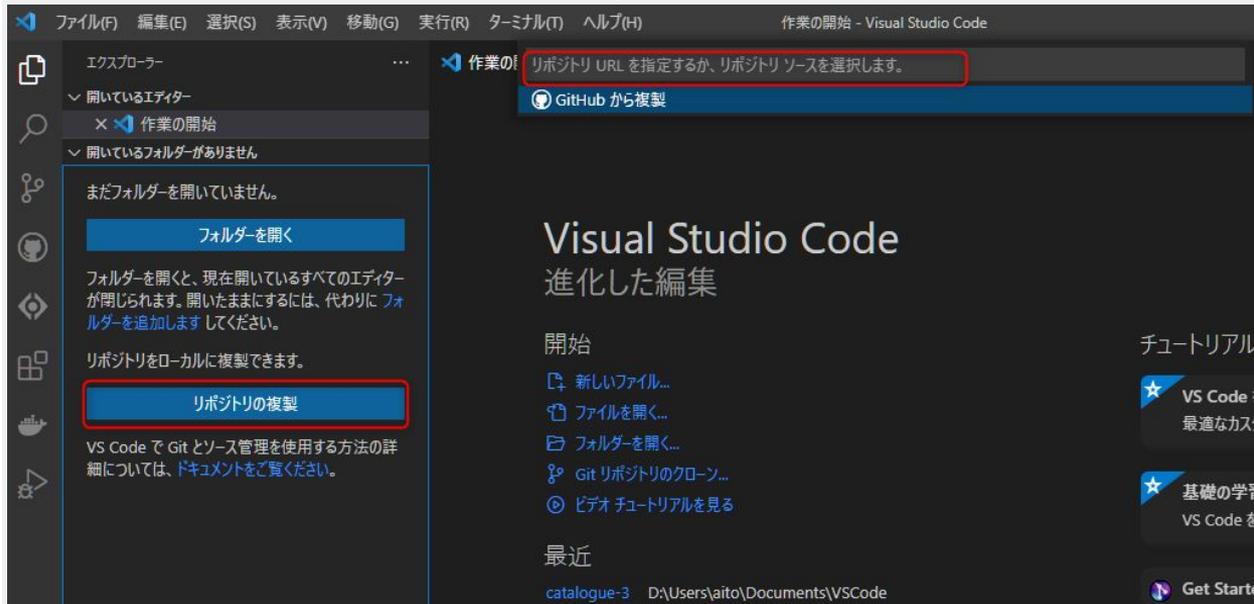
Requires GitLab Self-Managed 12.10 or later. [Check your New Relic CodeStream version.](#)

**GitLab Base URL**  
Please provide the Base URL used by your team to access GitLab.

**GitLab Personal Access Token**  
Provide a [personal access token](#) with the following scopes so that CodeStream can access your pull requests and issues: `api`.

# ソースコードを修正し てみる

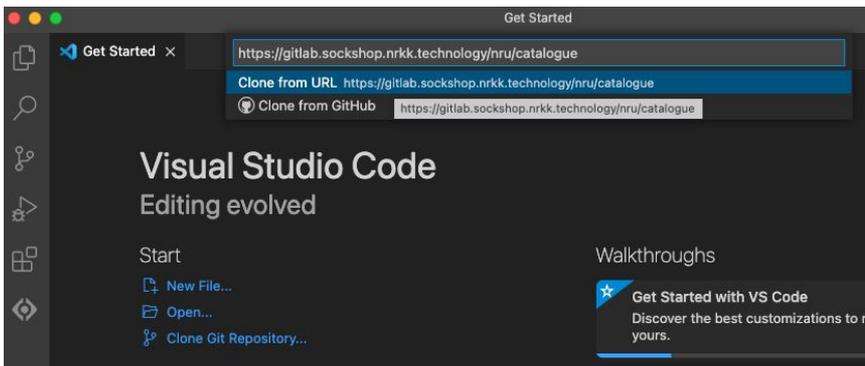
リポジトリをローカルに clone して開きます。



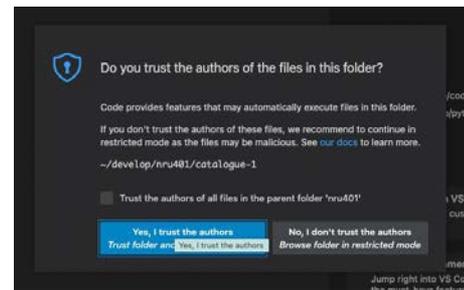
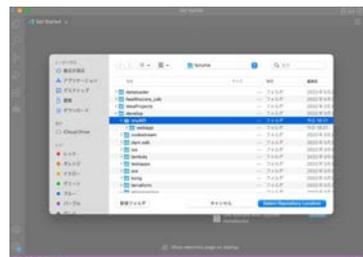
# GitLabからのプロジェクトクローン

1, VSCodeで Clone Git Repositoryを  
クリックし、URLを入力

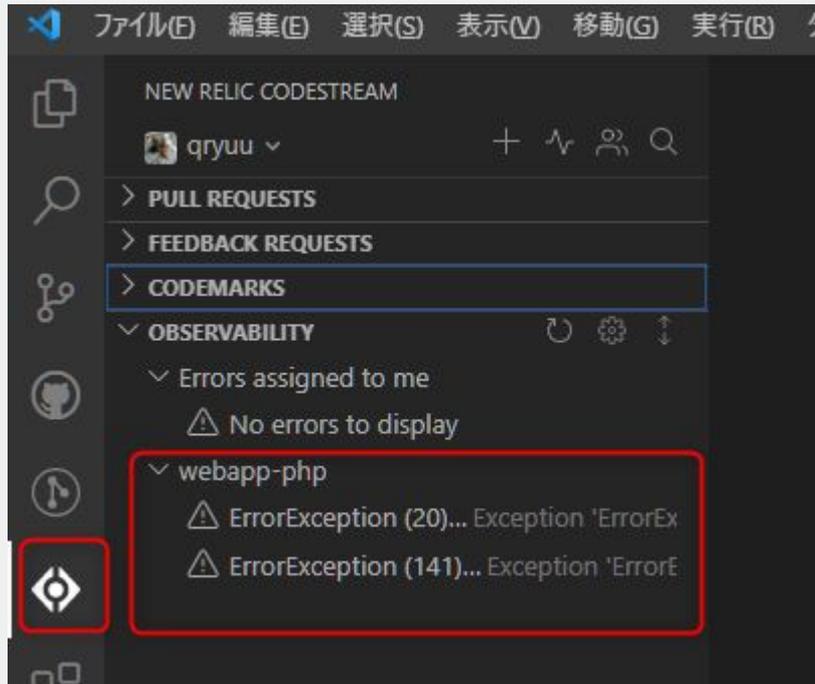
https://gitlab.sockshop.nrkk.technology/nru/catalogue



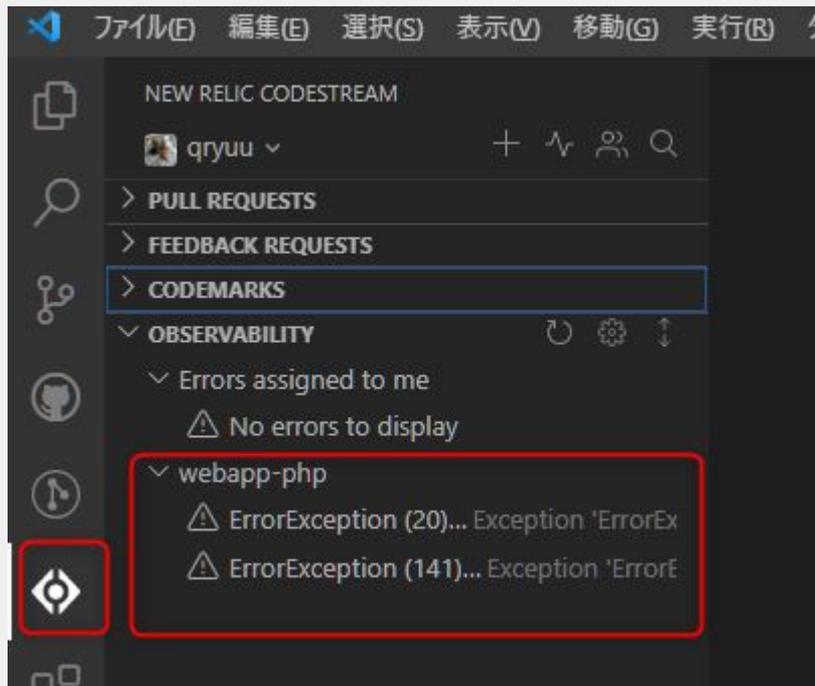
2, フォルダ選択等進みソースコードをクローン



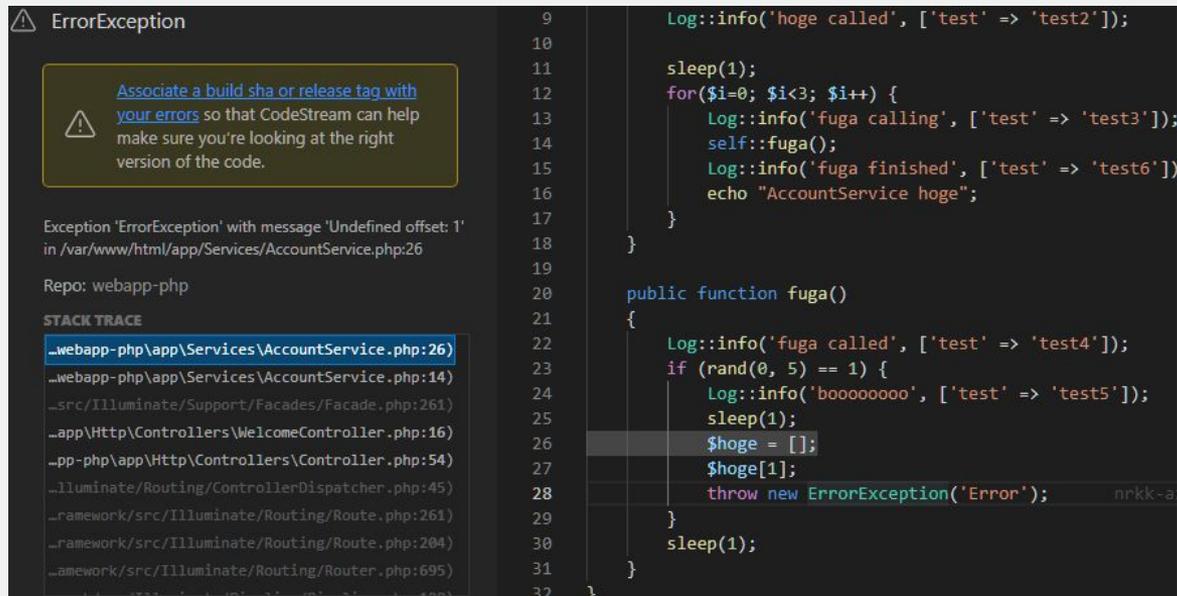
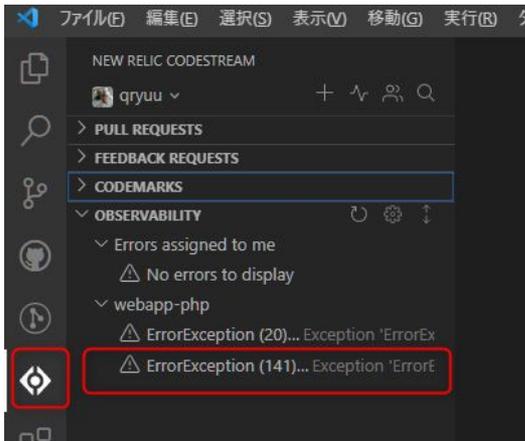
リポジトリを確認したら、Code Streamプラグインを確認します。IDE上でApplicationのErrorが確認できるようになります。



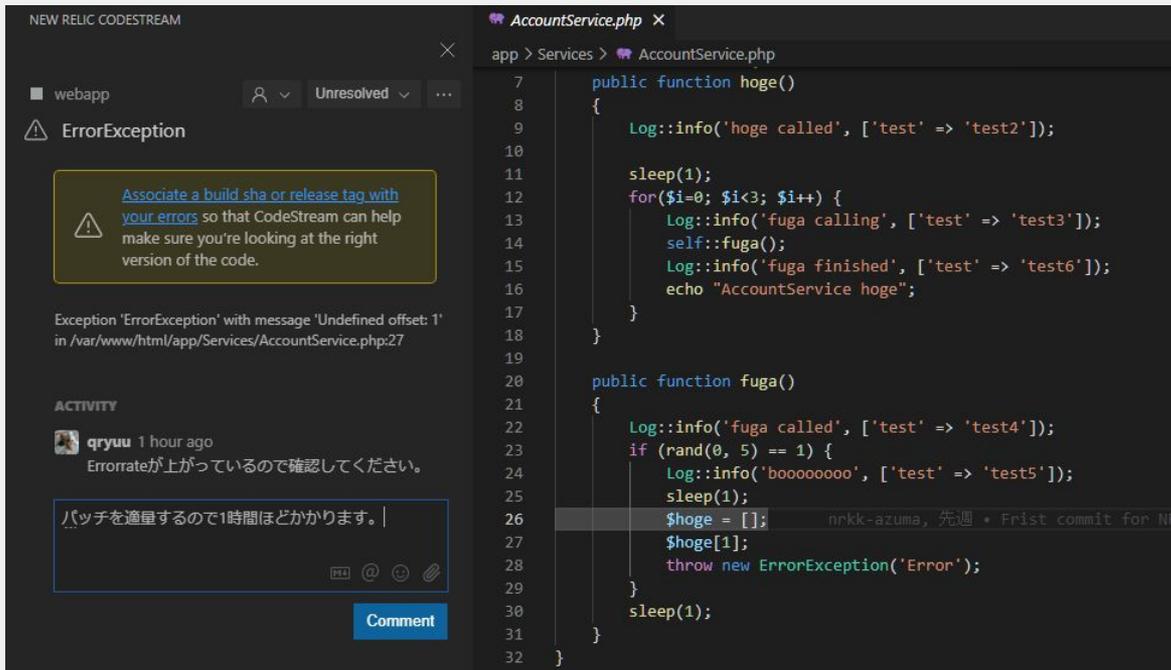
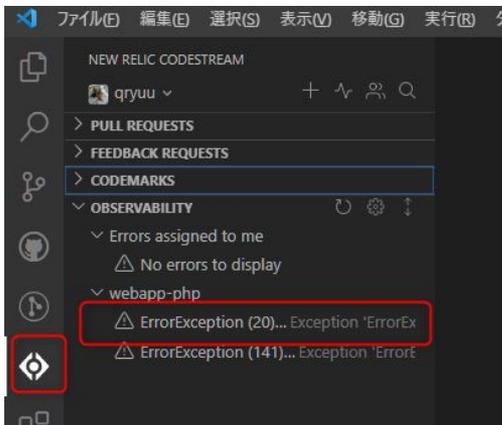
Errorを開きます。



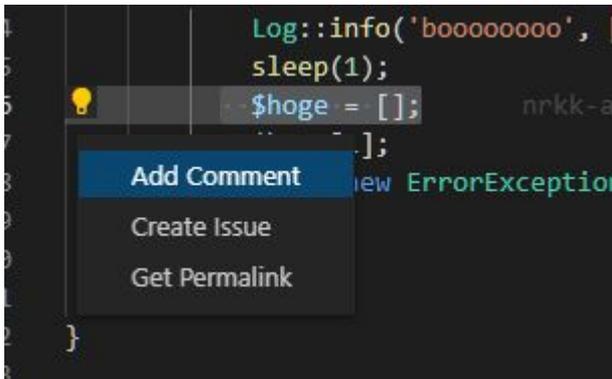
Errorを開き、スタックトレースを確認すると、関連するCode行に自動的に移動します。



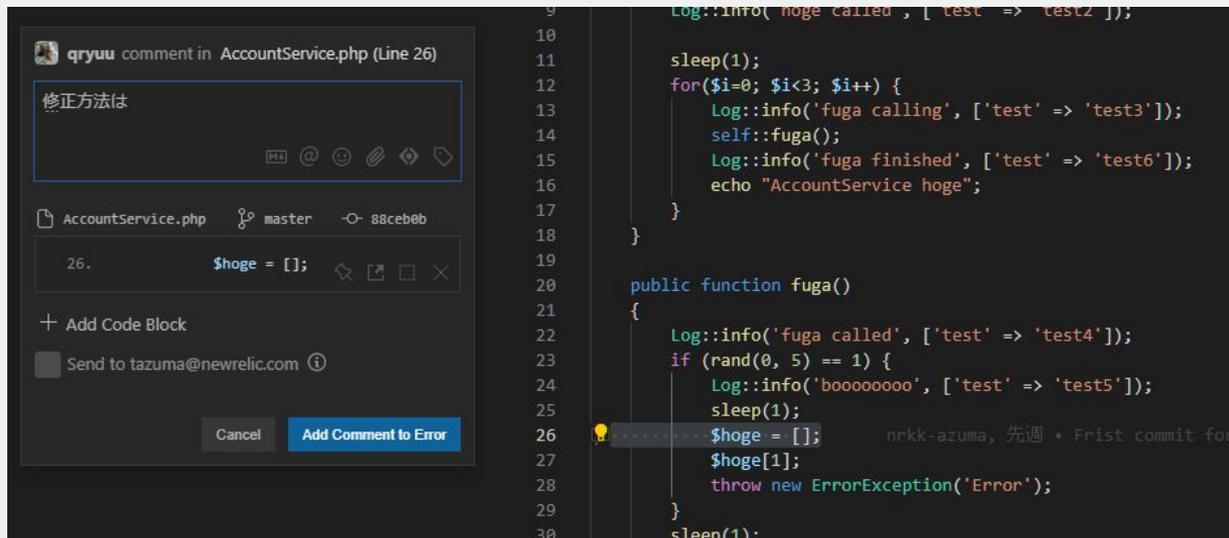
SREとのコミュニケーション。  
New Relicを使っているSREからのコメントもCode Streamで確認できます。



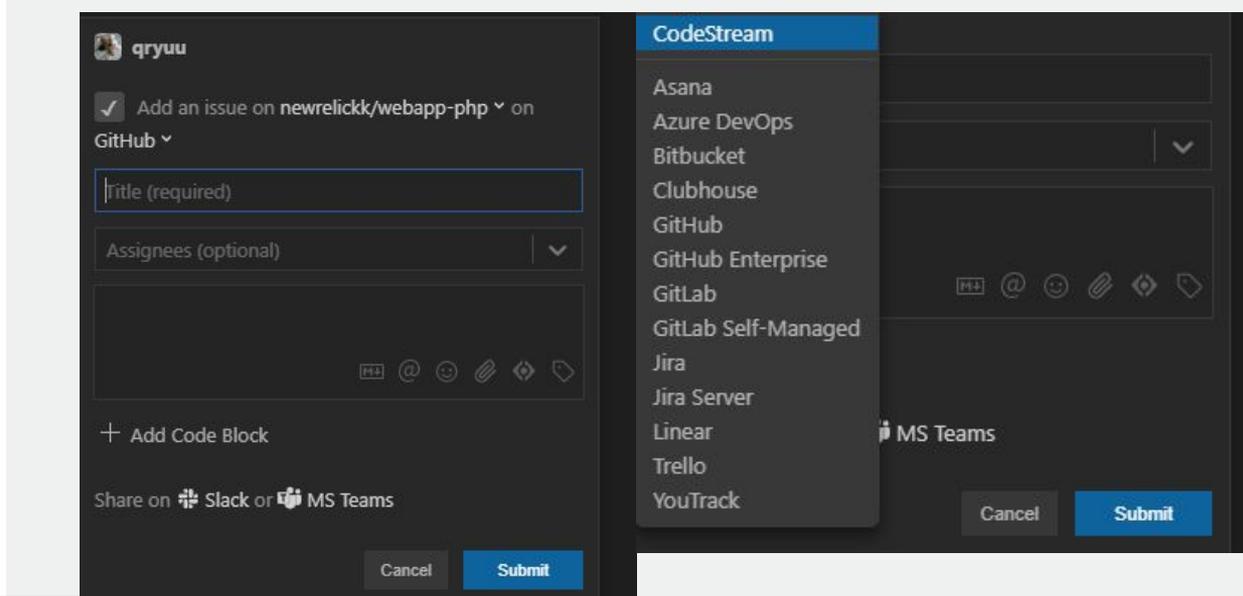
CodeStreamではCodeブロックに対してコメントやIssueを作成することができます。



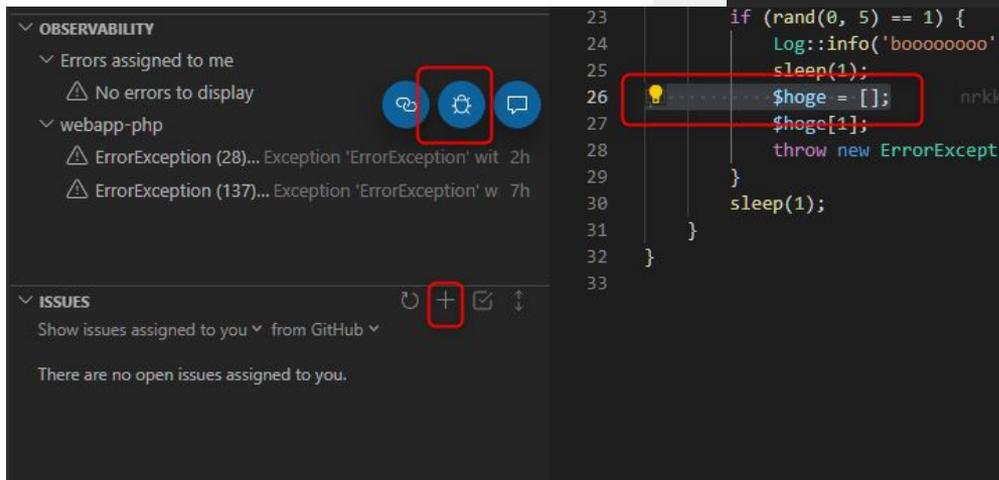
CodeStream自身がCodeを中心とした開発者同士のコミュニケーションツールになります。



CodeStreamから直接Issueを作成  
することができます。

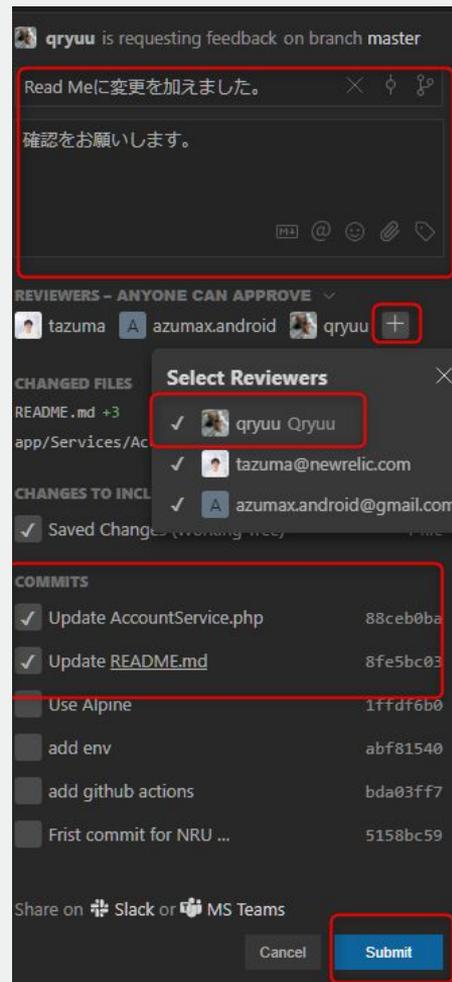


GitHubだけではなく様々なチケット管理ツールと連  
携する事ができます。



CodeStreamでは、リポジトリに  
プッシュする以前に簡単に Codeレ  
ビューを依頼する事ができます。  
CodeStream上で完結するこのレ  
ビューをFeedbackと呼びます。

FeedbackRequestでは、誰にどの  
ファイルを見てもらうのかを指定す  
る事ができます。



Feedbackでは追記や承認、差し戻しなどが行えます。

Read Meに変更を加えました。

+ 🍴 ↺ ⋮

qryuu made changes in `webapp-php` on `master` 3 minutes ago

**DESCRIPTION**  
確認をお願いします。

**REVIEWERS**  
 qryuu

**CHANGED FILES** ↓ ↑ 0 / 2

☰ ☰

▼ app/Services

- AccountService.php +1
- README.md +3

**COMMITTS**

- 📁 Uncommitted local changes
- 🕒 Update AccountService.php 88ceb0ba
- 🕒 Update README.md 8fe5bc03

Add Comment...

📎 @ 😊 ✎

Change Request (require for approval)

Add Comment

Feedbackで承認した変更はその  
ままPull Requestにすることができ  
ます。

PullRequestを実行する前にCode  
をpushしておいてください。

Read Meに変更を加えました。

+ 🍏 ↻ ...

**qryuu** made changes in `webapp-php` on `master` 3 minutes ago

**DESCRIPTION**  
確認をお願いします。

**REVIEWERS**  
**qryuu**

**CHANGED FILES** ↓ ↑ 0 / 2

- app/Services
  - AccountService.php +1
  - README.md +3

**COMMITTS**

- Uncommitted local changes
- Update AccountService.php 88ceb0ba
- Update README.md 8fe5bc03

Add Comment...

Change Request (require for approval)

Add Comment

Read Meに変更を加えました。

🔗 + ↻ ...

**qryuu** made changes in `webapp-php` on `master` 6 minutes ago

**DESCRIPTION**  
確認をお願いします。

**REVIEWERS**  
**qryuu**

**CHANGED FILES** ↓ ↑ 0 / 2

- app/Services
  - AccountService.php +1
  - README.md +3

**COMMITTS**

- Uncommitted local changes
- Update AccountService.php 88ceb0ba
- Update README.md 8fe5bc03

**ACTIVITY**

- qryuu** approved this feedback request just now

Add Comment...

🔗 🍏 🗑️ 🗒️

そのままCodeStreamで  
PullRequestのマージを行うことが  
できます。

Readmeを追記しました。 #1

qryuu wants to merge 1 commits into webapp-php:master  
from nru401-ito 5 minutes ago

Open

0 1 1 +1 -0

qryuu commented 5 minutes ago  
Author Collaborator

Changes re 23, 2022  
Author has been invited to collaborate on the repository.

Created from VS Code using CodeStream

Readmeを追記しました。 9bf3754

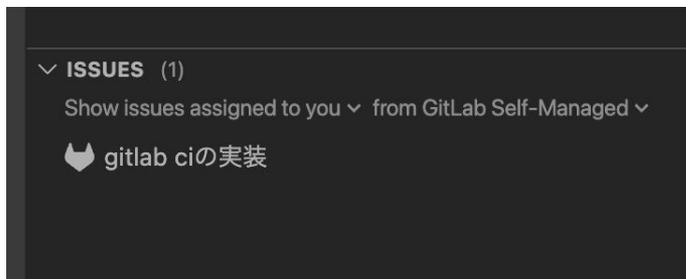
Add more commits by pushing to the nru401-ito branch on newrelick/webapp-php.

✓ This branch has no conflicts with the base branch  
Merging can be performed automatically.

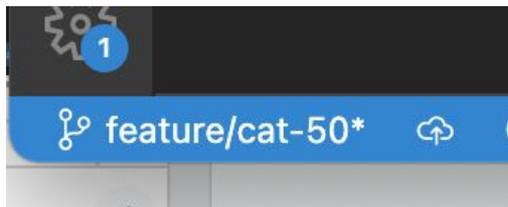
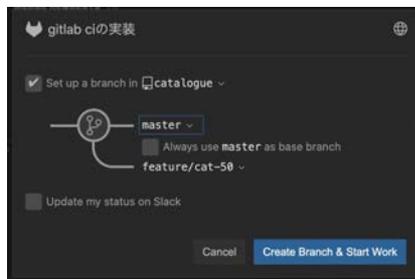
Squash and merge

# ソースコードの変更

## 1, ISSUESからチケット選択



## 2, Start Workをクリック

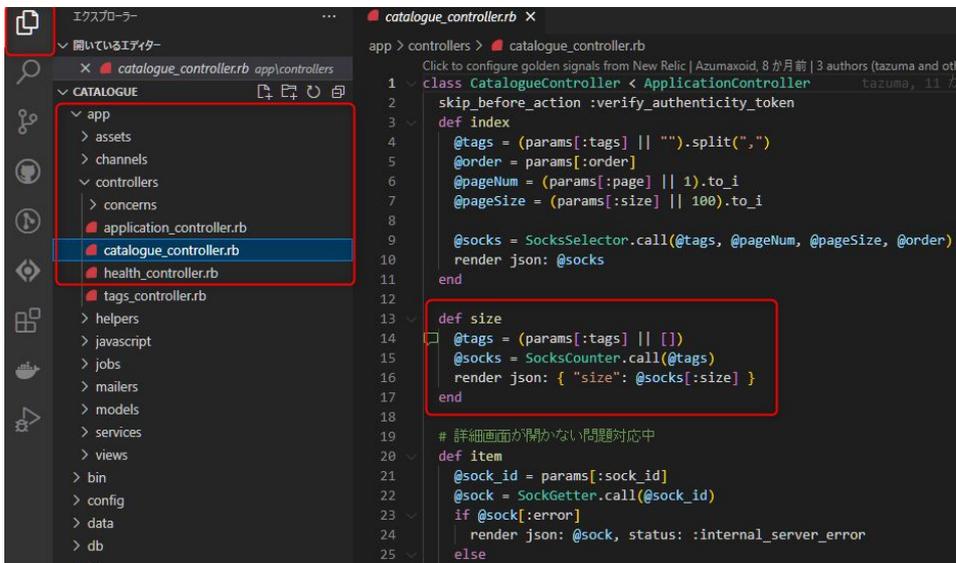


画面左下のブランチを確認

# ソースコードの変更

## 3, Errorの修正

app/controllers/catalogue\_controller.rbのメソッド"size"のソースコードを参考のように修正してコミットします。



```
1 class CatalogueController < ApplicationController
2   skip_before_action :verify_authenticity_token
3   def index
4     @tags = (params[:tags] || "").split(",")
5     @order = params[:order]
6     @pageNum = (params[:page] || 1).to_i
7     @pageSize = (params[:size] || 100).to_i
8
9     @socks = SocksSelector.call(@tags, @pageNum, @pageSize, @order)
10    render json: @socks
11  end
12
13  def size
14    @tags = (params[:tags] || []).split(",")
15    @socks = SocksCounter.call(@tags)
16    render json: { "size": @socks[:size] }
17  end
18
19  # 詳細画面が開かない問題対応中
20  def item
21    @sock_id = params[:sock_id]
22    @sock = SockGetter.call(@sock_id)
23    if @sock[:error]
24      render json: @sock, status: :internal_server_error
25    else
```

```
def size
  @tags = (params[:tags] || "").split(",")
  @socks = SocksCounter.call(@tags)
  render json: { "size": @socks[:size] }
end
```

# ソースコードの変更

## 4, ソースコードをPush



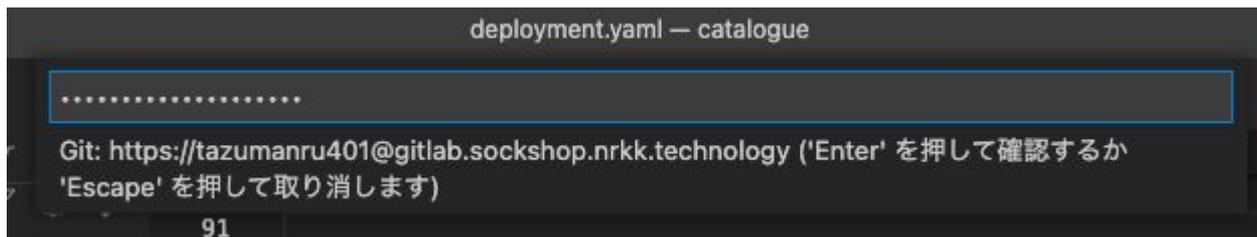
# 初めてのプッシュ時の認証

## 3, ダイアログにユーザー名を入力



```
deployment.yaml - catalogue
nru401user001
Git: https://gitlab.sockshop.nrkk.technology ('Enter' を押して確認するか 'Escape' を押して取り消します)
91
message: Unauthorized
```

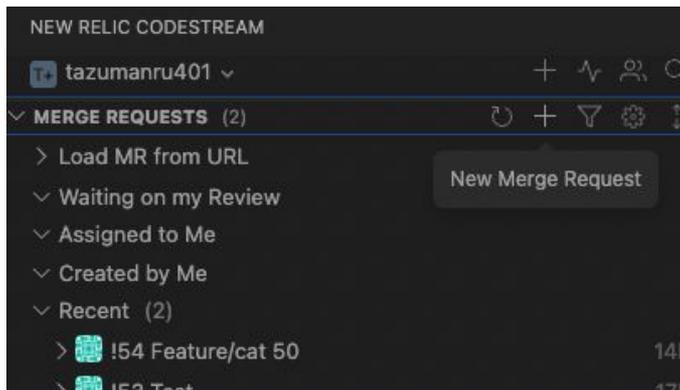
## 4, GitlabのAccess Tokenをペースト



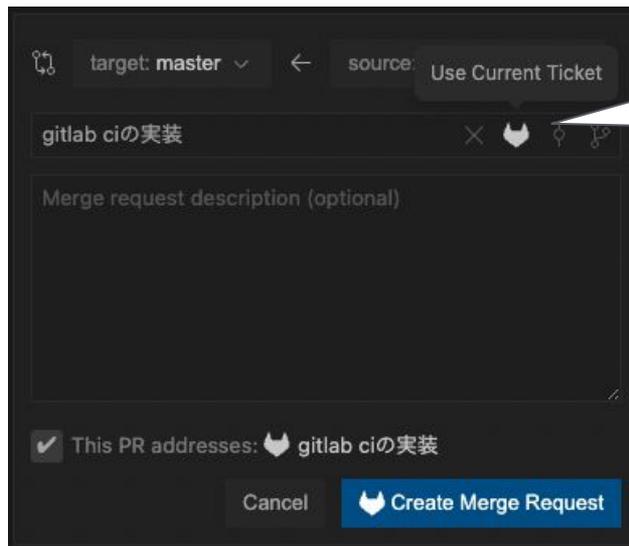
```
deployment.yaml - catalogue
.....
Git: https://tazumanru401@gitlab.sockshop.nrkk.technology ('Enter' を押して確認するか 'Escape' を押して取り消します)
91
```

# Merge Request (Pull Request)の作成

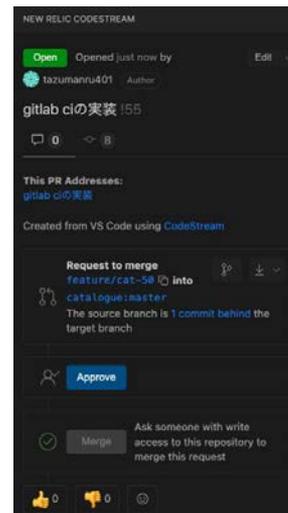
1, +アイコンをクリック



2, タイトルを入れて ボタンをクリック



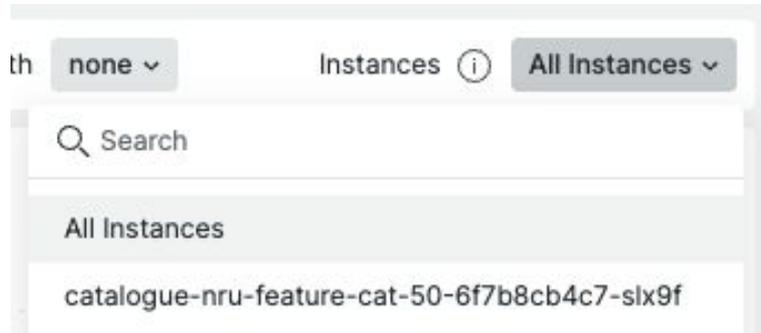
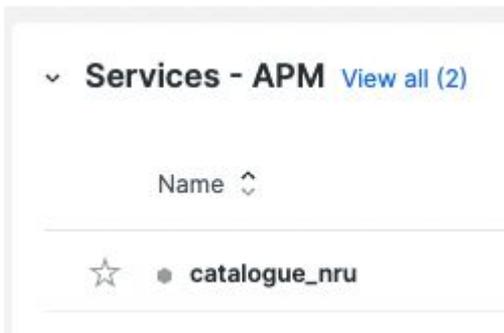
コミットメッセージやチケット名から簡単入力



# Merge Request作成後CICDパイプラインが実行

The screenshot displays the GitLab CI/CD interface. On the left, a sidebar menu includes 'Repository', 'Issues', 'Merge requests' (highlighted with a '1' badge), 'CI/CD', 'Security & Compliance', 'Deployments', and 'Packages & Registries'. The main content area shows a merge request titled 'gitlab ciの実装' with ID '!55', created 2 minutes ago. The right panel shows the pipeline details for 'gitlab ciの実装', indicating it is a 'Detached merge request pipeline #1'. The pipeline status is 'Running', and it shows two stages: 'Build' (with a 'docker-build' job) and 'Deploy' (with a 'deploy' job).

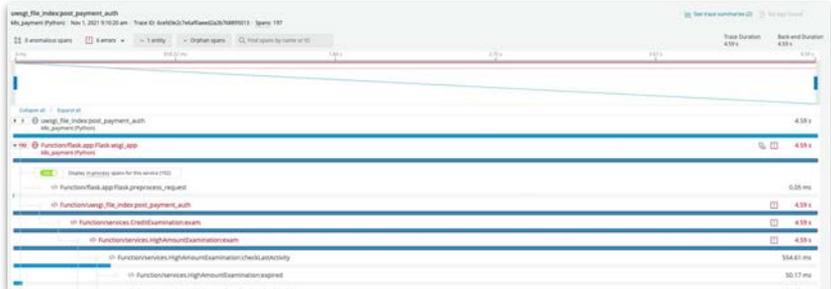
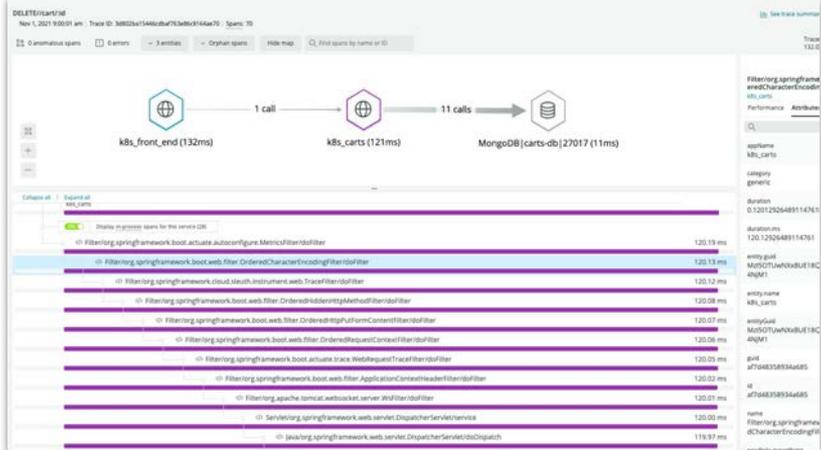
# New Relicでの確認



CD/CIパイプラインによりブランチ名付きのインスタンスが構築され、New Relicから確認できます。

# Errors in Boxとは





■ k8s\_orders | Unresolved

org.springframework.util.NestedServletException  
Request processing failed; nested exception is java.lang.IllegalStateException: Unable to create order due to unspecified ID

First seen: 7 minutes ago  
Last seen: 4 minutes ago

account: NRXX - Demo | accountID: 329505 | language: java

Occurrences: 7

Stack Trace

Error message: Request processing failed; nested exception is java.lang.IllegalStateException: Unable to create order due to unspecified ID

```

    org.springframework.web.servlet.FrameworkServlet.processRequest(FrameworkServlet.java:912)
    org.springframework.web.servlet.FrameworkServlet.doPost(FrameworkServlet.java:872)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:648)
    org.springframework.web.servlet.FrameworkServlet.service(FrameworkServlet.java:846)
    javax.servlet.http.HttpServlet.service(HttpServlet.java:728)
    org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:239)
    org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:165)
    org.apache.tomcat.websocket.server.WsFilter.doFilter(WsFilter.java:52)
    org.apache.catalina.core.ApplicationFilterChain.internalDoFilter(ApplicationFilterChain.java:232)
    org.apache.catalina.core.ApplicationFilterChain.doFilter(ApplicationFilterChain.java:165)
  
```

Key Attributes

- transactionName: WebTransaction/SpringControl... 100.0%
- transactionUriName: /OrderController/newOrder 100.0%

Feature attributes

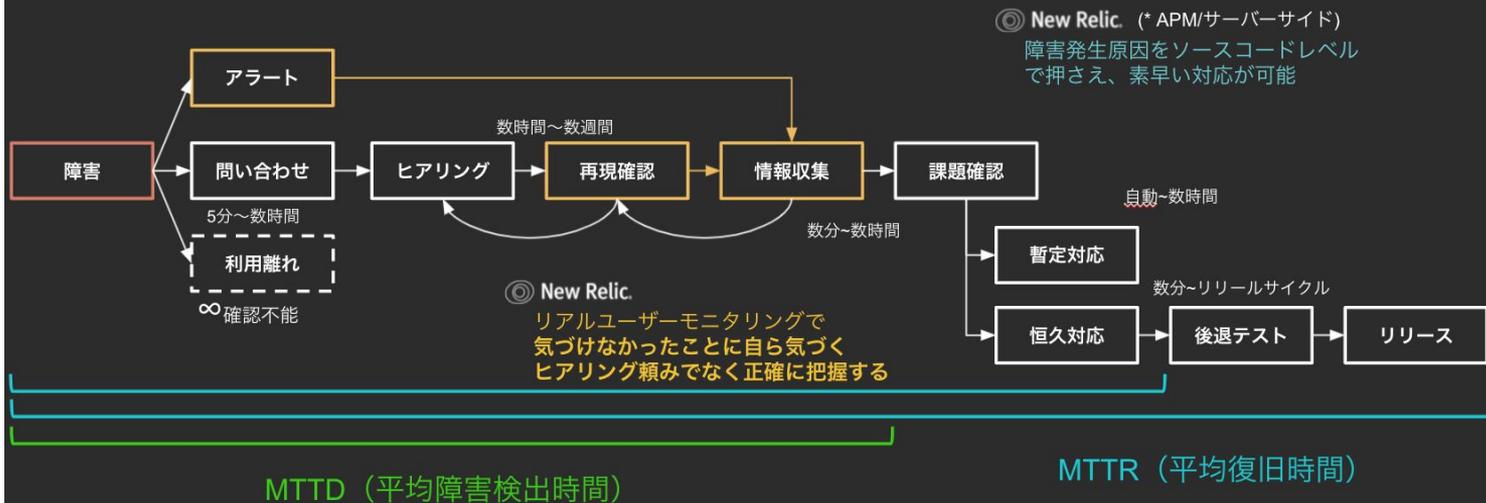
Key	Value

We couldn't find logs for this trace. This may be because your company hasn't enabled logs in context. Get started with log management

数週間→数分にするなど、自分たちで障害原因を理解できるようになった。

## 障害対応時のNew Relicの活用方法（参考時間）

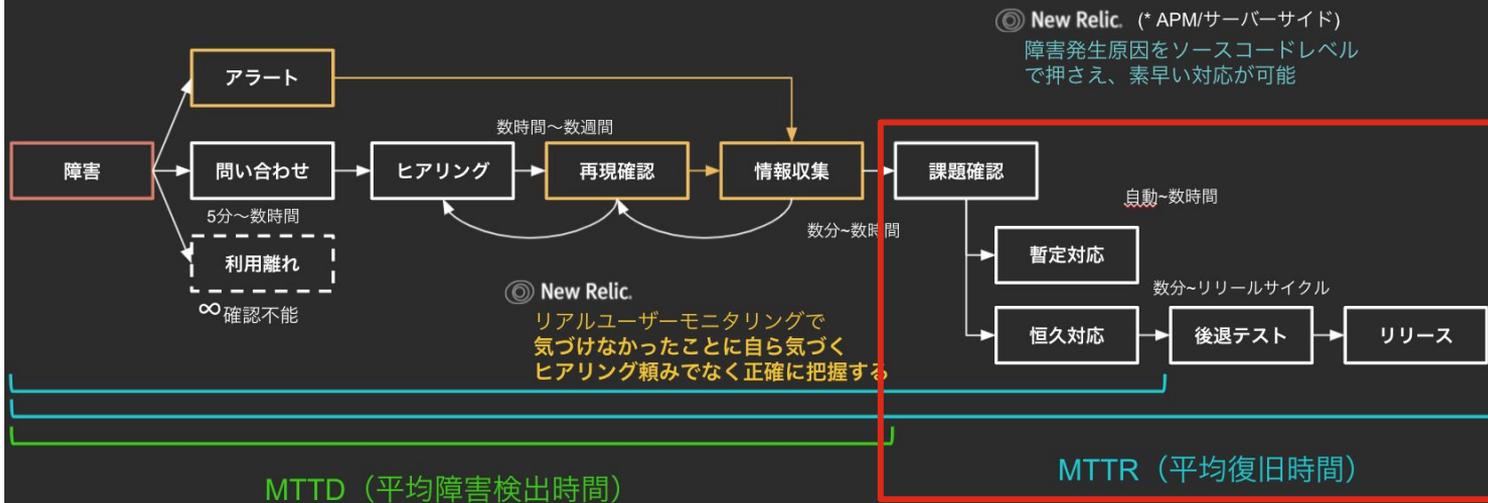
障害が発生してから対応を行うまでで最も時間のかかる「問題の特定」を、フロント/バック/インフラを含めE2Eで観測します。正常な状態においてもシステムがどうなっているのか、ユーザーがどう使っているのかをシステム全体を通して理解できるようにしておくことで、万一の障害対応に原因特定と障害対応だけでなく、障害に気づくまでの速度が上がり、状況を正確に把握できます。



数時間以上かかる“作業”がMTTRの方に残っている。

## 障害対応時のNew Relicの活用方法（参考時間）

障害が発生してから対応を行うまでで最も時間のかかる「問題の特定」を、フロント/バック/インフラを含めE2Eで観測します。正常な状態においてもシステムがどうなっているのか、ユーザーがどう使っているのかをシステム全体を通して理解できるようにしておくことで、万一の障害対応に原因特定と障害対応だけでなく、障害に気づくまでの速度が上がり、状況を正確に把握できます。



# New Relic One - Errors Inbox からの 開発ツールとの連携

The image illustrates the workflow of an error in New Relic One. It starts with the 'Errors Inbox' view, which lists error groups. A blue arrow points from an error entry to a detailed 'Occurrences' view. From there, another blue arrow points to the 'Stack Trace' section, which includes an 'Open in IDE' button. This button is shown in a separate IDE window, where the error details are displayed, including the 'NoMethodError' and the full stack trace.

**Errors Inbox View:**

- Workload: SockShop - JP
- Filter: error\_group.metadata.state = Unresolved
- Sort by: Occurrences
- ERROR GROUPS (14)
- ActiveRecord:RecordNotFound (772 occurrences)
- ActiveRecord:ConnectionNotEstablished (81 occurrences)
- Error connect ECONOMIESUSED (11 occurrences)
- Error HTTPError 406 (6 occurrences)
- works.weave.socks.orders.controllers.OrdersController\$PaymentDeclinedException (6 occurrences)

**Occurrences View:**

- Timestamp: Oct 20, 8:02:29 PM GMT+09:00
- URL host: catalogue
- URL path: /catalogue/00c7c3d6-af53-45a2-85c4-e4c32fa...
- Show only errors with stack trace:
- First of 1 of 34, Last

**Stack Trace View:**

```
Error message:
Couldn't find Sock with 'sock_id'=00c7c3d6-af53-45a2-85c4-e4c32faa88c6
...ocal/bundle/gems/activerecord-6.1.4.1/lib/active_record/core.rb:338:in `find'
/home/appuser/webapp/app/services/sock_getter.rb:9:in `call'
/home/appuser/webapp/app/services/service.rb:5:in `call'
/home/appuser/webapp/app/controllers/catalogue_controller.rb:23:in `block (2 levels) in <action_controller__>
...ck-6.1.4.1/lib/action_controller/metal/basic_implicit_render.rb:6:in `rescue in render_implicit_render'
.../bundle/gems/actionpack-6.1.4.1/lib/abstract_controller/base.rb:22:in `action'
```

# “CxO” にはこんな世界が待ってる

- 開発チームと運用チームの無駄な作業を削減し、**生産性の高い作業**にエンジニアリソースを投入できる。
- **エース級**の開発エンジニアがより**ビジネス貢献の高い領域**で活躍できる
- 質の高いサービスの維持と、**新機能/新サービスの市場へのリリースサイクルが早まる**
- **エンジニアの仕事環境の改善**

# ここがアツいよ！ CodeStream

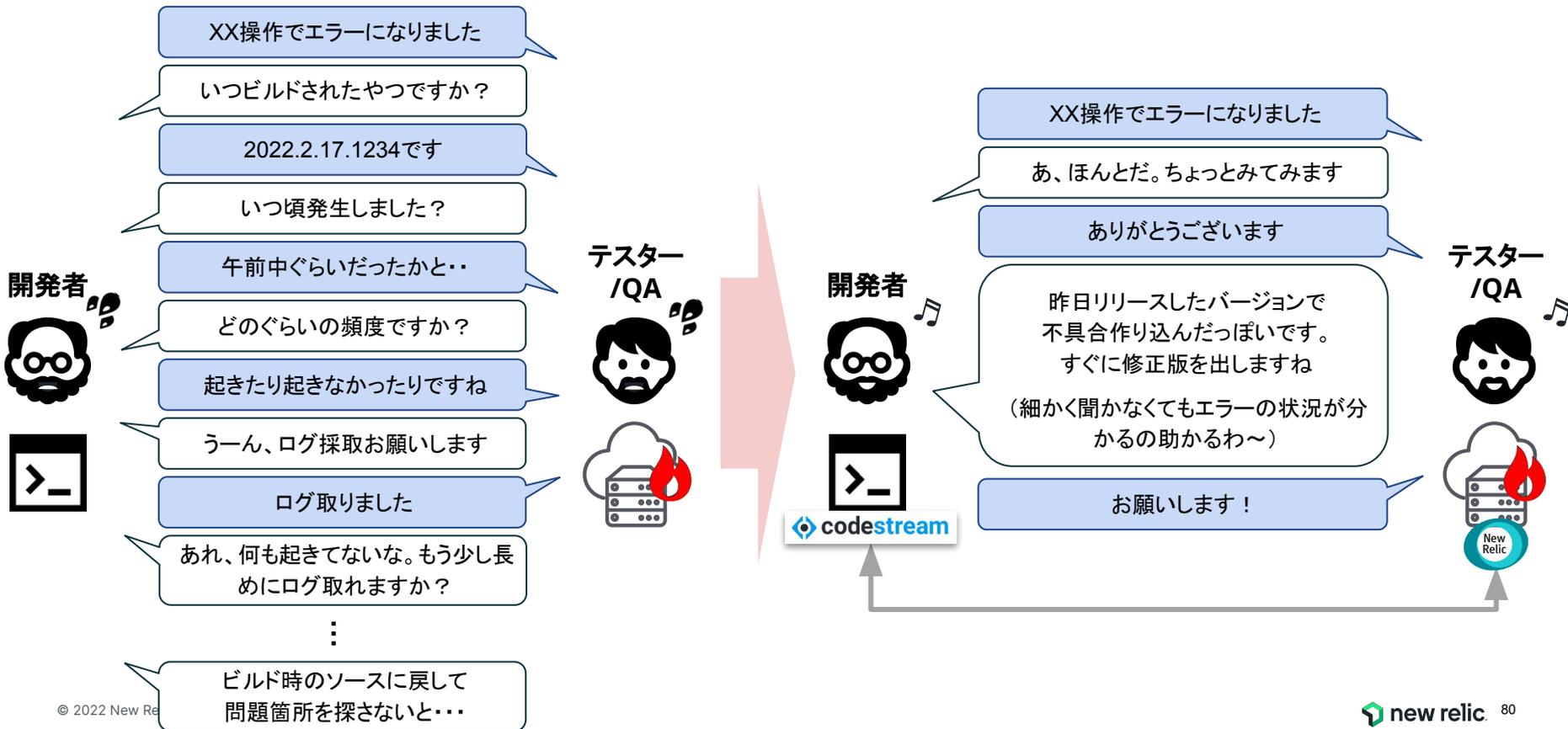
ここがアツい！① IDEですべてが完結！

ここがアツい！② 流れるようにコードで会話！

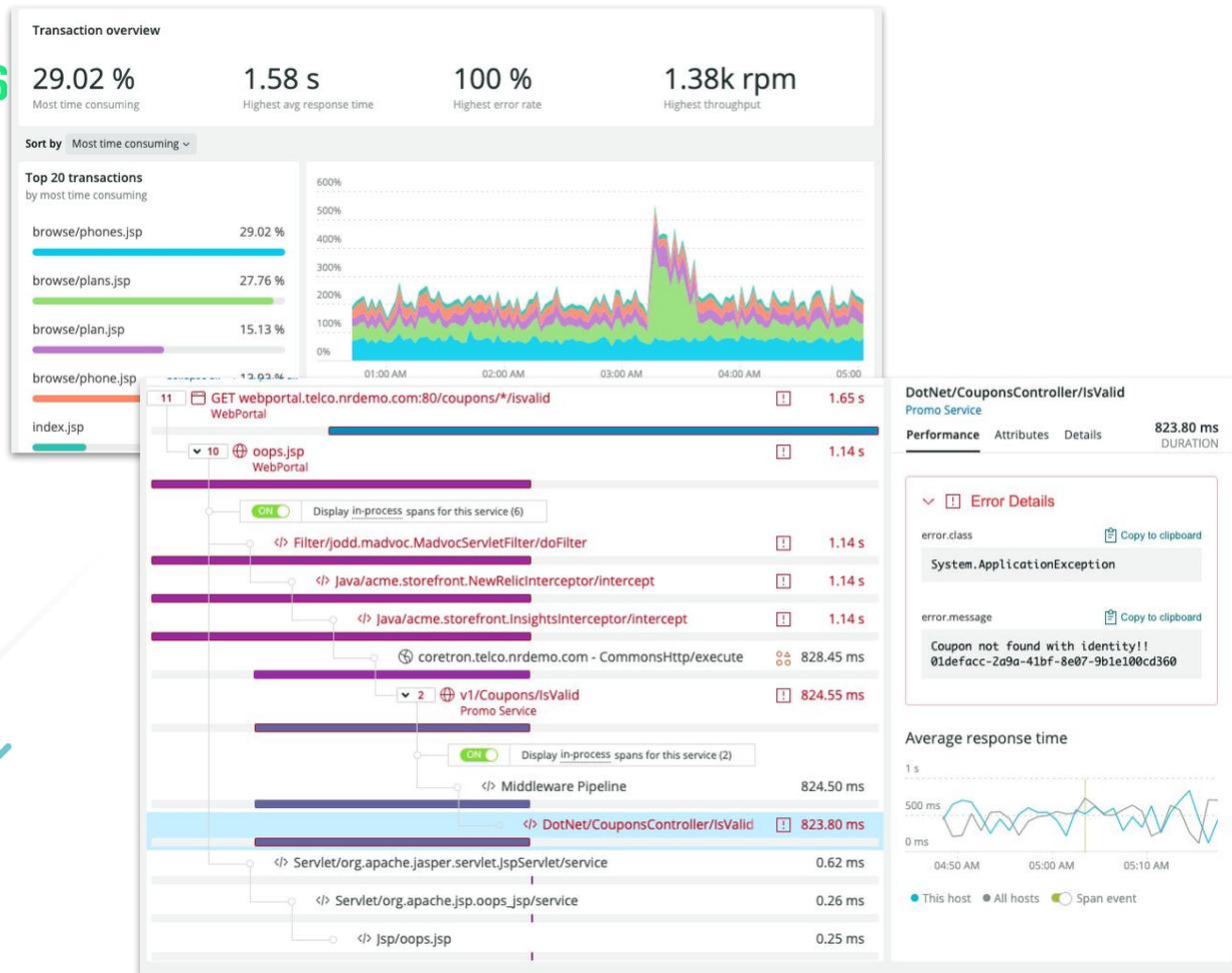
ここがアツい！③ アプリの問題コードがすぐわかる！



# IDEで問題コードを速攻で特定！



- ユーザー体験からインフラまでシステムの全容をリアルタイムに把握できる**オブザーバビリティ**(可観測性)を全てのステークホルダーに提供
- アプリケーションのトランザクション処理を観測し、ボトルネックやエラーをソースコードやDBクエリの詳細度で即座に特定



# システムのエラーから該当の ソースコードへ ダイレクトに到達

- 問題の詳細をすぐに把握。トラブルシュートに必要な資料採取やQA/Opsチームとのコミュニケーションを効率化
- 問題が発生しているコードを(リビジョン込みで)自動的に特定してIDEに表示。再現環境を作る手間をかけずに問題発生箇所を把握

The screenshot displays the CodeStream interface for a runtime error. The error details include:

- RuntimeError**
- Intended Error: error\_20220203
- First seen: 8 minutes ago
- Last seen: 8 minutes ago
- account: ksaito-demo
- accountid: 3074397
- language: ruby

An 'Occurrences' graph shows a single spike at 03:50 PM with 12 occurrences. The 'Stack Trace' section lists the following frames:

```

/app/app/controllers/settings_controller.rb:14:in `create'
...local/bundle/gems/actionpack-6.0.4.4/lib/abstract_controller/base.rb:195:in `process_action'
...le/gems/actionpack-6.0.4.4/lib/action_controller/metal/rendering.rb:30:in `process_action'
...ndle/gems/actionpack-6.0.4.4/lib/abstract_controller/callbacks.rb:42:in `block in process_action'
...e/gems/activerecord-6.0.4.4/lib/active_record/callbacks.rb:135:in `run_callbacks'
  
```

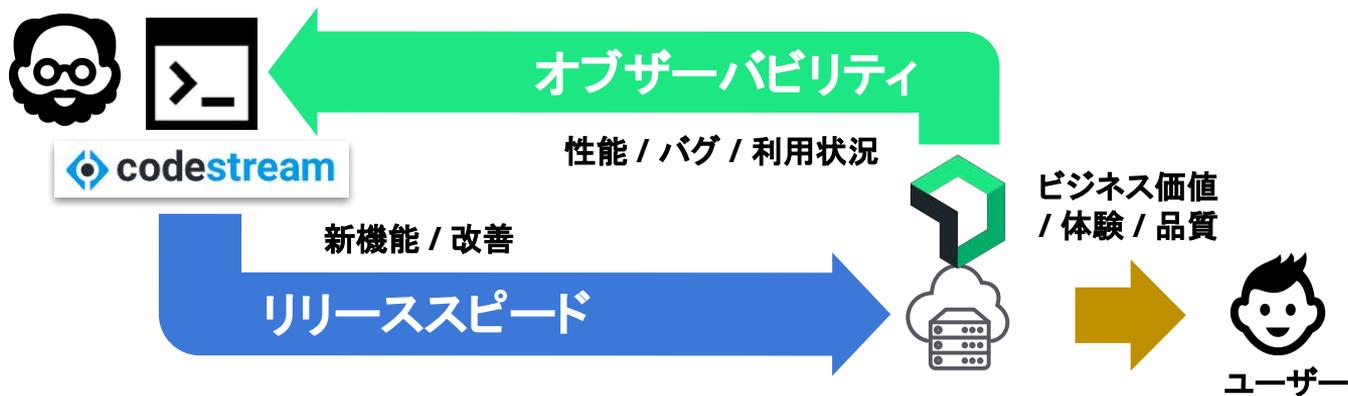
A red box highlights the 'Open in IDE' button in the stack trace, with a red arrow pointing to the IDE view below. The IDE view shows the source code for `settings_controller.rb` with the error location highlighted at line 15:

```

1 class SettingsController < ApplicationController
2   def index
3     @settings = Setting.all
4   end
5
6   def new
7     @setting = Setting.new
8   end
9
10  def create
11    @setting = Setting.new(setting_params)
12
13    if @setting.key.start_with?('error')
14      error_loop1("Intended Error: " + @setting.key)
15    elsif @setting.key == 'slow'
16      #Some comment...
17      100.times do
18        dummy = Setting.all
19      end
20    end
21
22    if @setting.save
23      redirect_to settings_path
  
```

# 開発者にオブザーバビリティを

スピーディかつ高品質に顧客価値を提供し続けるために、  
開発者自身がフィードバックを得て改善できるスキルが重要に



オブザーバビリティと改善の効率性によって次世代の開発者スキルを獲得

# Errors in Boxハンズオン

# APMからのError確認

ログインしたらAPMからWebAppを開きます。

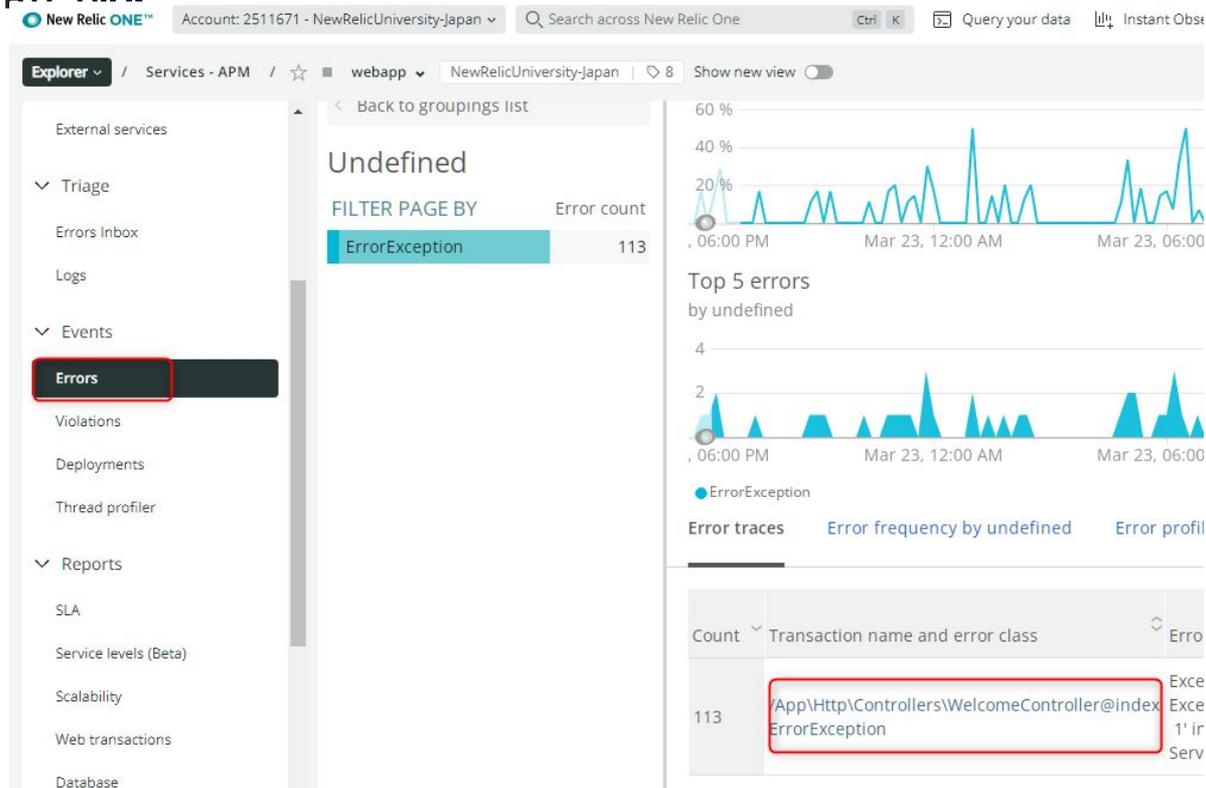
The screenshot shows the New Relic ONE interface. At the top, there is a search bar with the text "Search across New Relic One" and an account dropdown menu showing "Account: 2511671 - NewRelicUniversity-Japan". Below the search bar is a navigation menu with "Explorer" selected, and other options like "Browse data", "Dashboards", "Alerts & AI", "Errors Inbox", "APM", "Browser", "Infrastructure", and "Logs".

Under the "Explorer" section, there is a filter bar with the text "Filter by name, type, tags... (e.g. entityType = Host)". Below the filter bar is a sidebar with "Last viewed (3)" and "Saved views" sections. The "Last viewed (3)" section shows "All entities (369)", "Data gaps", and "Your system" with a sub-item "Services - APM (2)".

The main content area shows a list of services. The "Services - APM" section is expanded, showing a table with columns "Name", "Account", and "Er". The "webapp" item is highlighted with a red box. Below this, the "Browser applications" section is also expanded, showing a table with columns "Name", "Account", and "Spi". The "webapp" item is also highlighted with a red box.

# APMからのError確認

Errors を開き、どのようなErrorが発生しているかを確認します。



# APMからのError確認

続いてErrors inboxを開きます。  
Errors inbox では1件1件のError  
ではなく、ErrorをGroup化して、そ  
の傾向を表示し、より取り組むべき  
Errorをわかりやすく表示します。

The screenshot shows the New Relic APM interface. On the left sidebar, the 'Errors Inbox' menu item is highlighted with a red box. The main content area displays a list of error groups under the heading 'ERROR GROUPS (2)'. The first group is 'ErrorException' with a count of 67 occurrences and a line graph showing the trend over time. The second group is also 'ErrorException' with a count of 45 occurrences and a line graph. The 'Errors Inbox' menu item is highlighted with a red box.

# APMからのError確認

ErrorGroupをクリックするとそのErrorに誰が取り組むのか、Errorの内容、そして開発エンジニアとのコミュニケーションを管理することができます。

The screenshot displays the New Relic APM interface for an error group. On the left, the error details for 'ErrorException' are shown, including the message 'Undefined offset: 1' and a callout bubble labeled '担当者アサイン' (Assignee) pointing to the user icon. Below this is a line graph titled 'Occurrences' showing the frequency of the error over time. On the right, the 'Stack Trace' section lists the call stack, starting with 'ExceptionHandler' and ending with 'ExceptionHandler'.

webapp Unresolved  **担当者アサイン**

**ErrorException**  
Exception 'ErrorException' with message 'Undefined offset: 1' in /var/www/html/app/Services/AccountService.php

First seen about 7 hours ago  
Last seen 11 minutes ago

account: NewRelicUniversity-Japan accountid: 2511671  
language: php

Occurrences 68 OCCURRENCES

Timestamp: Mar 23, 5:35:21 PM GMT+09:00  
URL host: a893d5878921f4b6da29a2eb338449da-1960021783.a...  
URL path: /welcome

Stack Trace

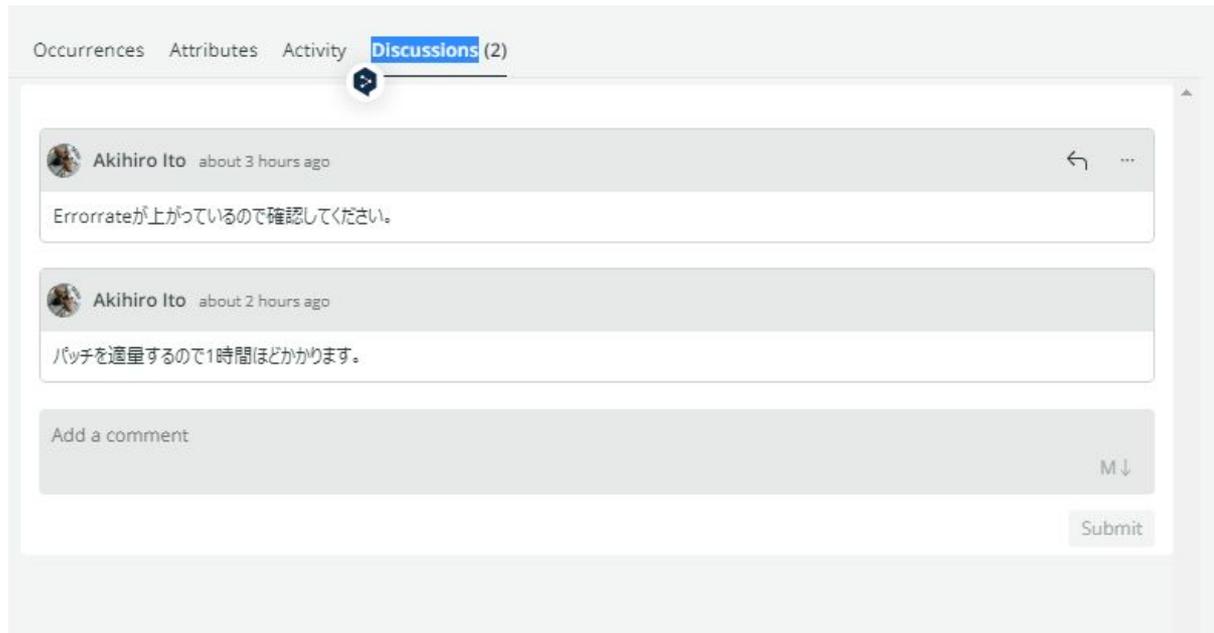
Exception "ErrorException" with message "Undefined offset: 1" in /var/www/html/app/Services/AccountService.php:27

- ...handleError called at /var/www/html/app/Services/AccountService.php (27)
- ...service::fuga called at /var/www/html/app/Services/AccountService.php (14)
- .../vendor/laravel/framework/src/Illuminate/Support/Facades/Facade.php (261)
- ...c called at /var/www/html/app/Http/Controllers/WelcomeController.php (16)
- .../html/vendor/laravel/framework/src/Illuminate/Routing/Controller.php (54)
- ...or/laravel/framework/src/Illuminate/Routing/ControllerDispatcher.php (45)
- ...r/www/html/vendor/laravel/framework/src/Illuminate/Routing/Route.php (261)
- ...r/www/html/vendor/laravel/framework/src/Illuminate/Routing/Route.php (204)
- ...www/html/vendor/laravel/framework/src/Illuminate/Routing/Router.php (695)
- ...w/html/vendor/laravel/framework/src/Illuminate/Pipeline/Pipeline.php (128)

Show all

# APMからのError確認

Discussionsではたの担当者やCodeStreamを使うエンジニアとコミュニケーションを取ることができます。



The screenshot shows the 'Discussions' tab in an APM tool interface. The tab is highlighted in blue and labeled '(2)'. Below the tab, there are two discussion entries by 'Akihiro Ito'. The first entry, posted 'about 3 hours ago', contains the text 'Errorrateが上がっているので確認してください。'. The second entry, posted 'about 2 hours ago', contains the text 'パッチを適量するので1時間ほどかかります。'. At the bottom of the discussion area, there is a text input field labeled 'Add a comment' and a 'Submit' button.

# APMからのError確認

Open IDEをクリックするとここからIDEを起動しそのまま原因となっているCodeを表示することもできます。  
そのままCodeStreamで開発やコード管理を行う事もできます。

Occurrences Attributes Activity Discussions (2)

Timestamp: Mar 23, 5:35:21 PM GMT+09:00 Show only errors with stack trace  First < 1 of 44 > Last

URL host: a893d5878921f4b6da29a2eb338449da-1960021783.a...  
URL path: /welcome

### Stack Trace

Copy to clipboard [</> Open in IDE](#)

Error message:  
Exception 'ErrorException' with message 'Undefined offset: 1' in /var/www/html/app/Services/AccountService.php:27

```
...:handleError called at /var/www/html/app/Services/AccountService.php (27)
...ervice::fuga called at /var/www/html/app/Services/AccountService.php (14)
...l/vendor/laravel/framework/src/Illuminate/Support/Facades/Facade.php (261)
...c called at /var/www/html/app/Http/Controllers/WelcomeController.php (16)
...html/vendor/laravel/framework/src/Illuminate/Routing/Controller.php (54)
...or/laravel/framework/src/Illuminate/Routing/ControllerDispatcher.php (45)
...r/www/html/vendor/laravel/framework/src/Illuminate/Routing/Route.php (261)
...r/www/html/vendor/laravel/framework/src/Illuminate/Routing/Route.php (204)
...www/html/vendor/laravel/framework/src/Illuminate/Routing/Router.php (695)
...w/html/vendor/laravel/framework/src/Illuminate/Pipeline/Pipeline.php (128)
```

Show all

# ぜひNew Relicをお試してください！

弊社サイトからサインアップ

<https://newrelic.com/signup>

With your new account, you get:

Perpetually free access

100 GB/month of free data ingest. 1 free full access user.  
Unlimited free basic users.

One data platform for all metrics, logs, events, and traces

Petabyte scale. Millisecond speed. Pennies per gigabyte  
(beyond free tier).

Easily visualize, analyze & troubleshoot your entire stack

APM, Infrastructure Monitoring, Digital Experience  
Monitoring, Applied Intelligence, and more.

 New Relic.

Free access to all of New Relic. Forever.

No credit card required.

NAME

e.g. Katherine Johnson

EMAIL

name@company.com

*By signing up, you're agreeing to [Terms of Service](#) and [Services Privacy Notice](#).*

Sign up

Have an account? [Log in](#)

# Learning Path

## Install

### インストールガイド

New Relic One へのサインアップやエージェントインストールの方法などのガイドを提供

[APM](#) / [Browser](#) / [Infrastructure](#) / [Logs](#) / [Mobile \(iOS/Android\)](#) / [AWS統合](#) / [Azure統合](#) / [GCP統合](#) インストール手順

▶インストールガイド

<https://newrelic.com/jp/blog/how-to-relic/new-relic-faststep-guide>

## 100

### 基本知識オンデマンドセミナー

New Relic One やオブザーバビリティに関する基礎知識を座学にて学習

[NRU Practitioner](#) オブザーバビリティ入門

[NRU 101](#) New Relic One 入門

▶オンデマンドセミナー

(ページ作成中)

## 200

### New Relic One 機能解説動画

New Relic One に含まれる3つの主要機能に含まれる54の機能群を動画で説明

[NRU201](#) Telemetry Data Platform

[NRU202](#) Full Stack Observability

[NRU203](#) Applied Intelligence

▶主要機能解説動画

<https://newrelic.com/jp/resources/data-sheets/nru201>

## 300

### ハンズオントレーニング (エンジニア一般向け)

New Relic One を実際に操作し、主要機能を利用できる状態にするためのトレーニング

[NRU 301](#) アプリケーションとインフラ性能観測の基本

[NRU 302](#) ダッシュボード開発とNRQLの基本

[NRU 303](#) SLI/SLO設計の基本

[NRU 304](#) AIOps とアラート設計の基本

▶開催スケジュール

<https://newrelic.com/jp/events>

## 400

### ハンズオントレーニング (開発者向け)

New Relic One の開発者向け機能を利用できる状態にするためのトレーニング

[NRU 401](#) CodeStream による DevOps を想定したエラー分析対応の基本

▶開催スケジュール

<https://newrelic.com/jp/events>



お疲れさまでした。

ご質問があればチャットにご記入ください  
アンケートにご協力お願いいたします

