

Amazon Q Developer in GitHubで 静的サイト作ってみた。

2025/09/19

村上 萌々香





アジェンダ

ツール説明

今回の動機

静的サイト仕様

Amazon Q Developer in GitHub導入

開発手順

実演

感想



ツール説明1: Amazon Q Developer

- Amazon Q Developer は、生成人工知能 (AI) を活用した会話型アシスタントで、AWS アプリケーションの理解、構築、拡張、運用に役立ちます。
- 統合開発環境 (IDE) で使用すると、Amazon Q はソフトウェア開発を支援します。Amazon Q では、コードに関するチャット、インラインコードの完了、新しいコードの生成、セキュリティ上の脆弱性のスキャン、言語の更新、デバッグ、最適化などのコードのアップグレードと改善を行うことができます。

([公式ドキュメント](#)より)



+



ツール説明2: Amazon Q Developer in GitHub

- GitHubのIssueやPR上でAmazon Q Developerをシームレスに使えるようにするApp
- 2025年5月からプレビュー版が使用可能に
- AWSアカウントが無くても使用可能
- 各種コマンドが用意されている
 - /q dev: 開発実施
 - /q review: PRレビュー
 - /q transform: javaバージョン更新



ツール説明3: GitHub Pages

- **GitHubのリポジトリから直接ウェブサイトを公開できる
静的ホスティングサービス**
- 無料アカウント：全体公開リポジトリでのみPagesが作成可能
→サイトも全体公開
- 有料アカウント：限定公開サイトが作成可能

今回の動機

- 入社時の開発環境構築手順をちゃんと管理していない
 - 毎年去年のslackチャンネルから手順書（mdファイル）を探して渡していた
 - 都度更新もされないなので、毎年見たことがあるエラーの対応をやりがち
- **GitHub Pagesで管理・確認出来たら便利そう**
- **色々なREADMEや1年生の全体会資料とかも管理出来たらいいな**

静的サイト仕様1: 概要

- GitHub Pagesベースの社内ドキュメントサイト
 - 目的: 社内手順書・仕様書の一元管理
 - アクセス: 社内に限定 (privateリポジトリ) ※今回は公開サイトでベース作成
 - 管理対象: mdファイル形式のドキュメント
- Q. なぜmdファイル形式?
- A. フロント知識が無くても、ある程度決まった様式で書けるから

静的サイト仕様2：運用方法

- ページ追加フロー

1. mdファイルを作成
2. ファイル配置: 指定フォルダにmdファイルを配置
3. 設定更新: config.jsonにページ設定を追加
 - 新規ページのページ配置構造/見出し名など
4. 自動デプロイ(GitHub Actions)
 - mdファイルをhtml形式に変換
 - GitHub Pagesに反映



Amazon Q Developer in GitHub導入

1: Amazon Q developer有効化

マーケットプレイスから有効化

<https://github.com/marketplace/amazon-q-developer>

- 今はプレビュー版なので無料？
- 月毎に呼び出し制限があるらしい

Review your order

Account:  FMs-murakami ▼

Order summary

Amazon Q Developer

Free

Free



\$0 / month

Total amount

Due today

Prorated for Aug

Billing information

Please confirm your information once.



FMs-m
Personal

First name *

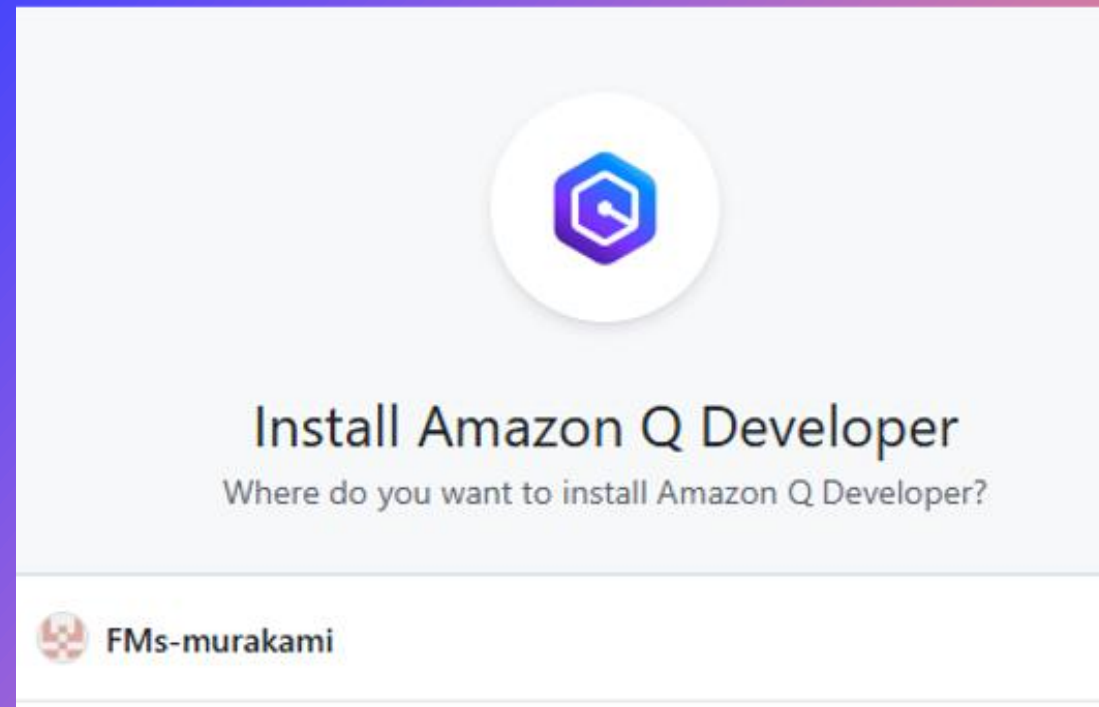
Address (Street, Apt. No.) *

Address line 2 (Apartment, Suite, etc.)

City *

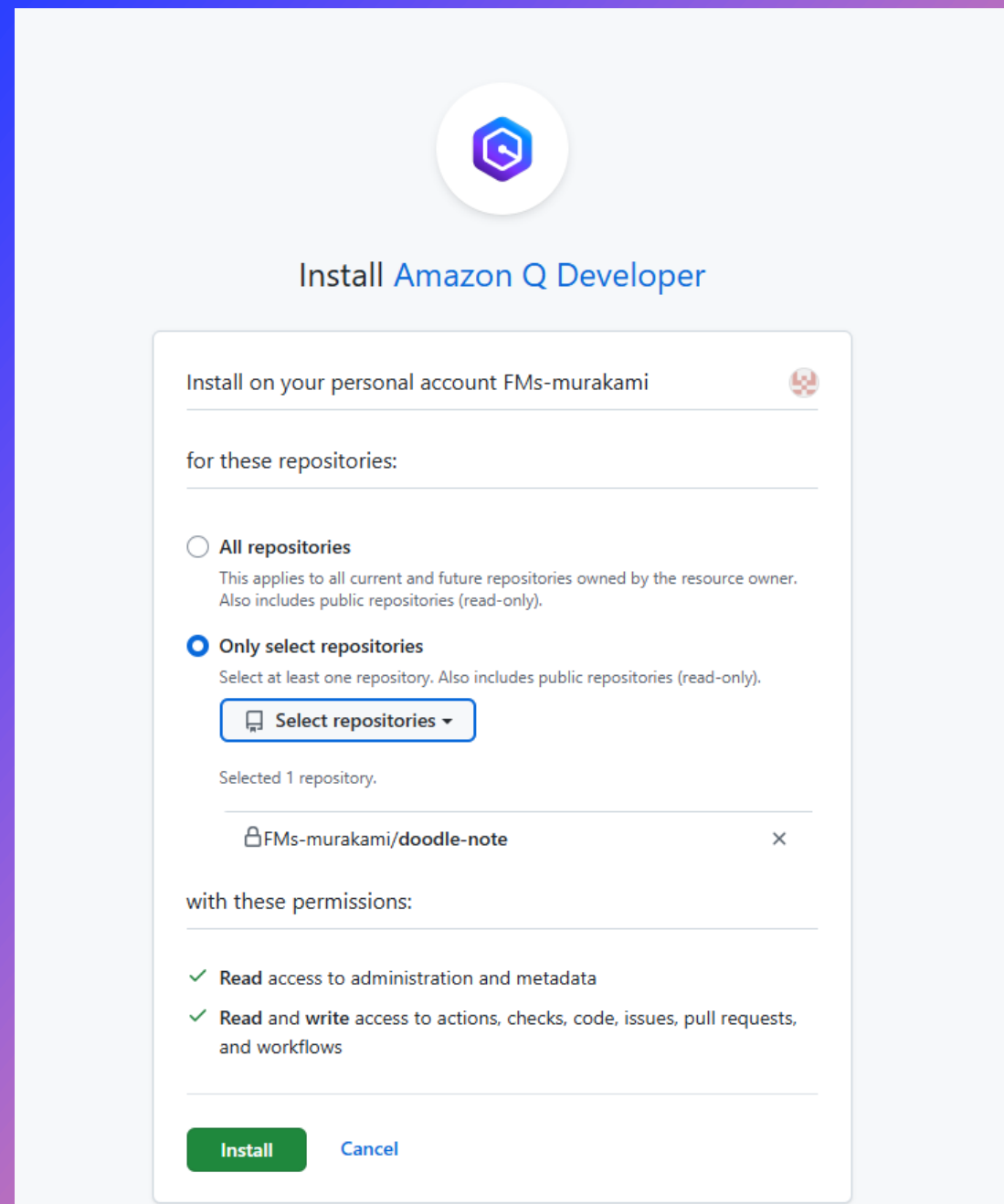
2: アカウント設定

- 個人アカウント/団体で設定可能
- 今回は個人アカウントに設定



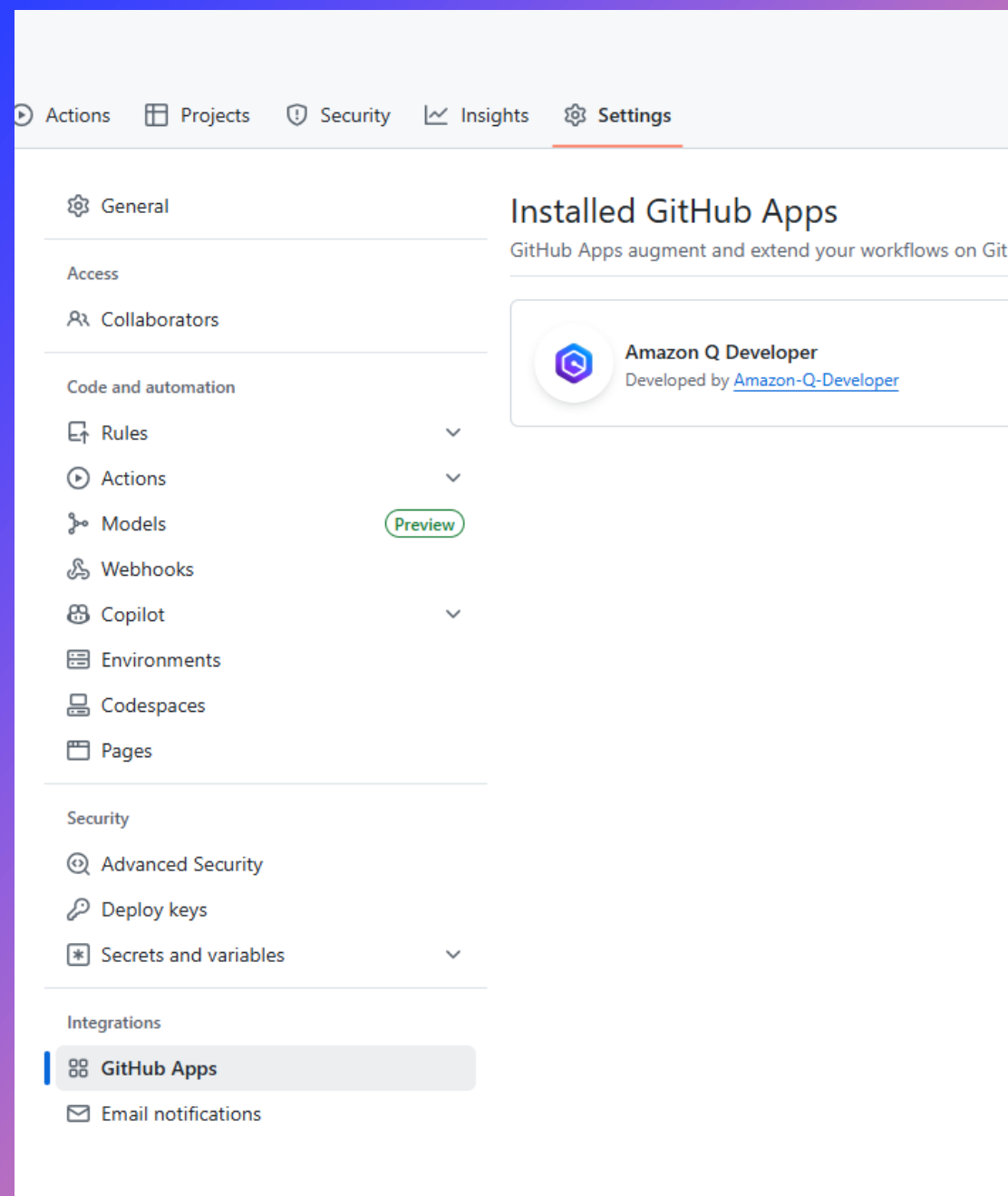
3: 対象リポジトリ設定

- Amazon Qを有効化するリポジトリを選択
- 個人アカウントに紐づくリポジトリから選択可能
- 今回は新規のリポジトリに設定



4: 設定確認

1. 対象リポジトリの設定ページに移動
2. GitHubAppページにAmazon QがあればOK



+

●

○

開発手順

1: Issue作成

指示は具体的に、細かくステップに切り分けて記載

- 大まかにステップに切り分け
 1. GitHub Pagesのセットアップ
 2. GitHub Actionsでの自動デプロイ設定
 3. mdファイルのHTML変換実装
 4. UI整備（サイドバー等）
- 各ステップごとにIssueを作成
- 今回はClaudeにまとめてもらった

Phase1: GitHub Pagesセットアップ #1

Open



FMs-murakami opened now

全体仕様

GitHub Pagesベースの社内ドキュメントサイトを作成

- 目的: 社内手順書・仕様書の一元管理
- アクセス: 社内に限定（privateリポジトリ）
- 管理対象: mdファイル形式のドキュメント
- ページ追加フロー
 - a. ファイル配置: 指定フォルダに.mdファイルを配置
 - b. 設定更新: config.jsonにページ設定を追記（mdファイルパス/見出し名/tagなど）
 - c. 自動デプロイ (GitHub Actions)

このissueではベースとなるGitHub Pages設定を整備します。

1. リポジトリの作成と初期設定

- リポジトリの説明文に「社内ドキュメント管理サイト」を記載
- README.mdの初期版を作成（プロジェクト概要を記載）

2. 基本ディレクトリ構造の作成

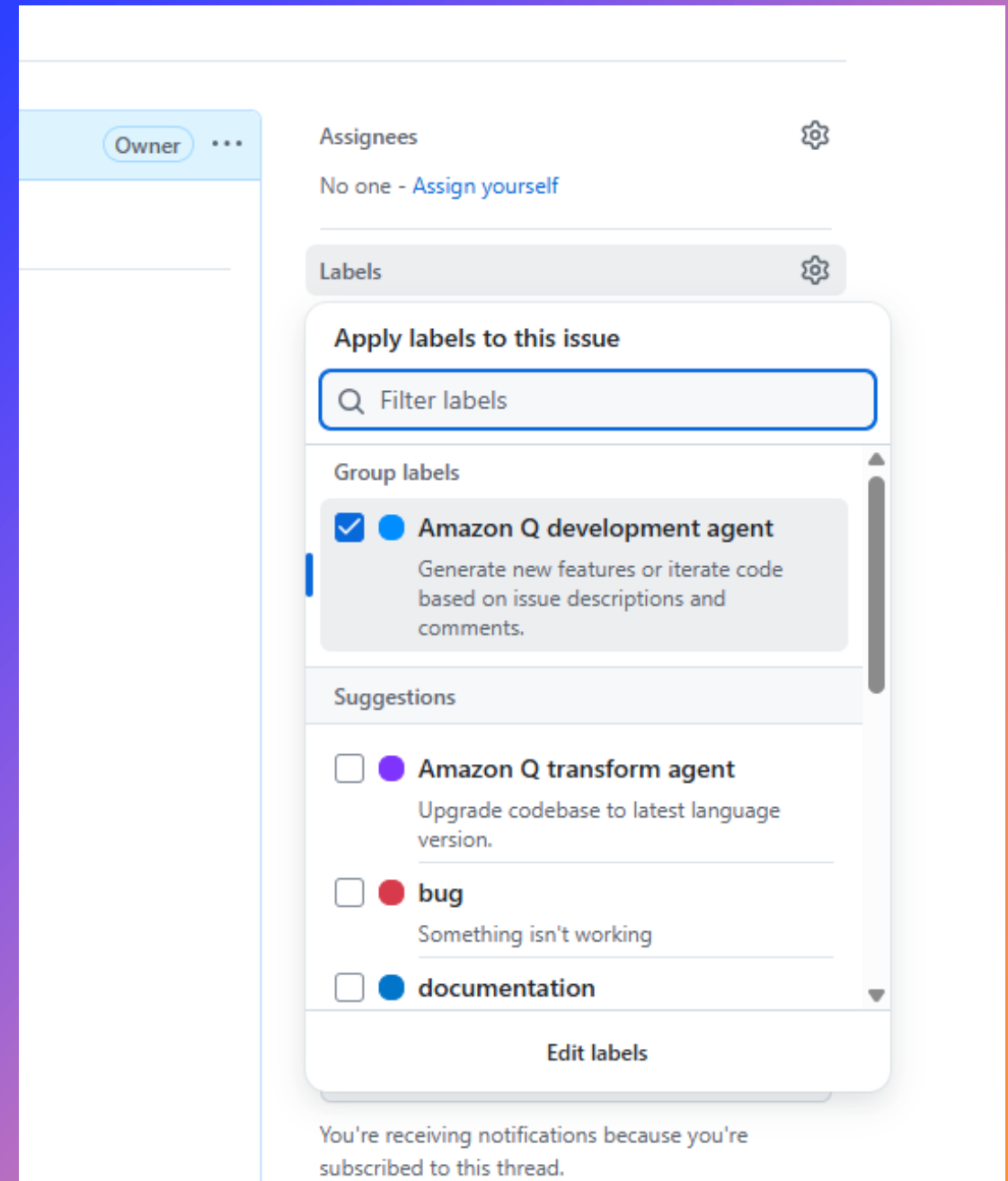
- docs/ フォルダを作成（Markdownファイル格納用）
- config/ フォルダを作成（設定ファイル用）
- .github/workflows/ フォルダを作成（後のActions用）
- assets/css/ フォルダを作成（スタイルファイル用）
- assets/js/ フォルダを作成（JavaScriptファイル用）

2: Issueのラベルに Amazon Qを設定

これだけでOK。

後の実装はAmazon Qが勝手にやってくれる

- 大体5~10分待てば完了
- 今何やっているかを都度コメントで教えてくれる
- 実装が終わったらPRまで作ってくれる



3: PR確認

実装が終わったらPRまで作成してくれる

- 詳細が英語なので気力で読む
- PRレビューまでAmazon Qがやってくれる
- PRのコメントに「/q review」と打つとレビュー依頼が可能

Phase1: GitHub Pagesセットアップ #2



amazon-q-develo... wants to merge 1 commit into `main` from `Q-DEV-issue-1-1755007091`



Conversation 12



Commits 1



Checks 0



Files changed 12



amazon-q-developer bot commented 5 minutes ago

This pull request implements the initial setup of a company internal documentation site using C... include:

1. Core Site Structure:

- Created basic HTML, CSS, and JavaScript files for the documentation site
- Set up configuration files for site management
- Implemented directory structure for documentation storage

2. Key Features:

- Real-time search functionality
- Tag-based content filtering
- Responsive design support
- Japanese language optimization
- Testing framework and deployment checklist

3. Documentation and Quality:

- Added comprehensive documentation for setup and deployment
- Implemented test files for structure and functionality verification
- Created detailed implementation summary and deployment guides

4. Theme and UI:

- Implemented a clean, professional design theme
- Added support for Japanese fonts (Noto Sans JP)
- Created card-based content display system

困ったこと

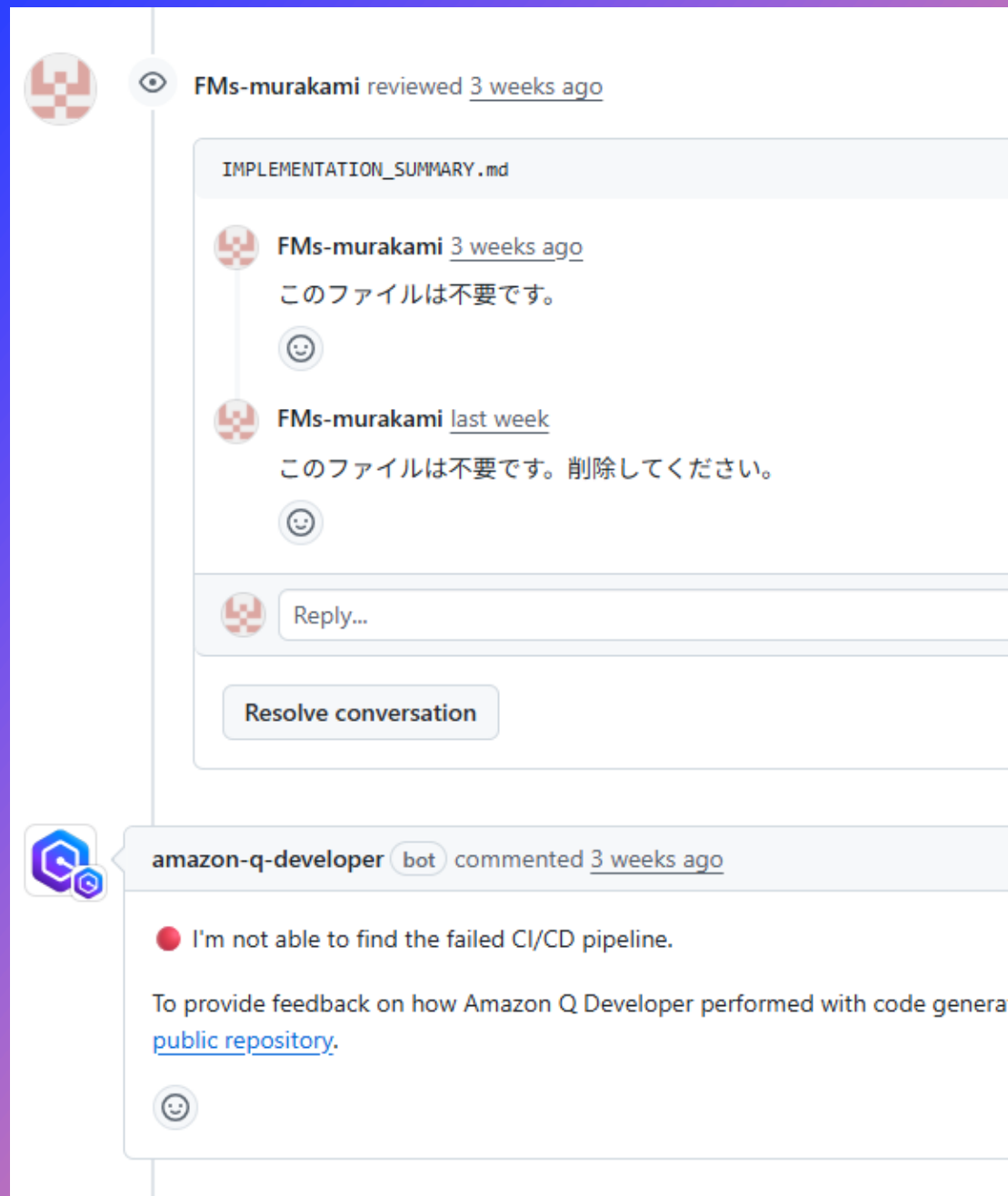
PR, Issueのコメントに反応してくれない

生成物に対して「ここを直してほしい」と

コメントしたが、以下エラーが出て対応してくれない

I'm not able to find the failed CI/CD pipeline.

- 設定不足 or プレビュー版だから？



The screenshot shows a GitHub conversation thread. At the top, a user named 'FMs-murakami' reviewed a file named 'IMPLEMENTATION_SUMMARY.md' 3 weeks ago. Below this, there are two comments from 'FMs-murakami': one from 3 weeks ago stating 'このファイルは不要です。' (This file is not needed.) and another from last week stating 'このファイルは不要です。削除してください。' (This file is not needed. Please delete it.). Below these comments is a 'Reply...' input field and a 'Resolve conversation' button. At the bottom of the thread, a bot named 'amazon-q-developer' commented 3 weeks ago with the message: 'I'm not able to find the failed CI/CD pipeline. To provide feedback on how Amazon Q Developer performed with code generation, please provide feedback on how Amazon Q Developer performed with code generation in our [public repository](#).' The bot's message includes a red dot icon and a smiley face icon.

基本運用

1. Issue作成
2. ラベルにAmazon Q設定
3. PRをざっと確認
4. 開発ブランチにpush
 - GitActionが起動→GitHub Pagesに反映される
5. 動作確認出来たら、そのままリリース
6. 不具合調整のIssueを作成（1に戻る）

この繰り返しで開発を進めた

表示微調整 #13

Open



FMs-murakami opened 1 minute ago

メニューはサイドバーに移動

- /環境構築/windows.md
- 環境構築/mac.md

上記のような配置の場合、サイドバーでは「環境構築」に折り畳みでページリンクが2つ入っている状態
ページに飛んだ時、メニューの表示がおかしくなる。サイドバーで常に表示されるようにしてほしい。

hogehoge

上記のようなコマンドラインの場合、灰色の四角に内容を書いて、右上にコピーアイコンを配置する

Create sub-issue



FMs-murakami added **Amazon Q develop...** 1 minute ago



amazon-q-developer bot 1 minute ago · edited by amazon-q-developer

▼ Progress Updates

- I need to explore the repository structure to understand the codebase and identify how the sidebar menu is currently implemented. Let me start by examining the workspace directory.
- I can see this appears to be a documentation site with markdown files and scripts for building. Let me check the file and the sidebar script to understand the current implementation.
- This appears to be a main landing page. Let me check the sidebar script and the page template to understand how navigation is implemented for individual pages.
- Now let me check the page template to see how the sidebar is integrated into individual pages, and also understand the current styling.

+

●

○

実演

感想

- 実装がGitHub(Issue→PR→GitHub Action→GitHub Pages)だけで完結していて、実装進行はとても楽だった
- PRのコメントで微調整できないのが不便だった
- まだ日本語に対応しきれていない？
 - Issueの日本語は理解してくれたが、出力（PR詳細やコメント）は英語固定
- フロント側の知識を付けないと、サイト実装のPRを確認するのが難しい
 - AIの実装を過信せず、ちゃんと目を通してからリリースする
- ドキュメント管理はBacklogのドキュメント機能使った方が良い

参考

- AWSブログ : Amazon Q Develop in GitHub(プレビュー)
 - <https://aws.amazon.com/jp/blogs/news/amazon-q-developer-in-github-now-in-preview-with-code-generation-review-and-legacy-transformation-capabilities/>
- 使ってみたブログ
 - <https://dev.classmethod.jp/articles/q-developer-integration-github-preview/>

ご清聴ありがとうございます

質問コーナー

Amazon Qの質問でも、
静的サイトの質問でも、
どちらでもOKです

