Definitions and Examples for Developer Needs (DN) and Relevant Information (RI) Categories

Introduction

This document provides the definitions and examples of Relevant Information (RI) categories and their corresponding Developer Needs (DN) classes, which you will use during the annotation process. Please refer to these definitions when categorizing sentences to ensure consistency and accuracy in your annotations.

1 Developer Needs (DN) Classes

The RI categories are associated into Developer Needs (DN) classes, which represent broader categories of information that developers seek. Table 1 shows how each RI category corresponds to a specific DN class. Use this table as a reference to understand the broader context of each RI category.

DN_id	**Developer Need**	**Definition**
EH	Error Handling	The developer identifies an issue within their code that causes unexpected behavior or failures and seeks solutions to diagnose and resolve the error.
FI	Functionality Implementation	The developer aims to design and implement new features or enhancements within their project using GitHub Actions to automate workflows and processes.
OR	Orientation	The developer looks for advice, best practices, or recommendations on how to proceed with a particular task or decision within their project, using GitHub Actions.
LE	GHA Learning	The developer is looking to acquire knowledge and understanding of GitHub Actions, requiring documentation, tutorials, or examples to learn how to effectively use its features and capabilities.
II	Insufficient Implementation	The developer finds that their current implementation falls short of the desired functionality or specifications, necessitating further enhancements or modifications.
IN	Incompatibility	The developer's code functions correctly in their local environment but encounters issues or fails to execute as expected when run within GitHub Actions.
MI	Migration	The developer seeks to transition their continuous integration and continuous deployment (CI/CD) processes from another platform to GitHub Actions, ensuring compatibility and functionality during the migration.
AS	Alternative Solution	The developer has an existing solution in place but is interested in exploring different methods or tools that might offer better performance, efficiency, or simplicity.

Table 1: Detailed Taxonomy of Developer Needs

2 Relevant Information (RI) Definitions

Each RI category is associated with a specific type of information need that developers express in their posts. The table below (Table 2) provides the detailed definitions of each RI category. You should use these definitions to guide your classification of the sentences.

RI_id	Relevant Information	Definition	Example
EH1	Where does the error occur?	This includes specific locations in the code such as functions, steps, jobs, stages, or modules where the error manifests.	Deploying using Github actions in Digital ocean docker username and password error
EH2	When does the error occur?	This details the timing of the error, whether it happens at execution, after a certain period, or at the end. It also includes whether the error is constant or intermittent.	The 'strange' thing is that sometimes the test passes and sometimes it doesn't.
EH3	Were there recent changes in the code?	This involves describing any recent changes made by the developer to the code before the error appeared. These changes could be relevant to understanding the cause of the error.	When I try to switch projects via -CODE- it errors with the error: -CODE-.
EH4	What actions precede the error?	This includes the specific actions or parts of the code that are executed just before the error occurs.	While connecting to AWS and retrieving the audio as a pydub.AudioSegment works, accessing the metadata fails.
EH5	What is the erroneous behavior?	This describes the incorrect behavior exhibited by the code that indicates an error.	Github actions can not run the config.sh file in the ubuntu 20.04
EH6	What is the cause of the error?	This provides a description of the possible reason or cause behind the error.	This fails in Github Actions because Github Actions can not download the private repository without some token.
EH7	What is the error message?	This includes the exact error message received, which can help in diagnosing the problem.	Here is the full error I have been getting: -CODE-.
EH8	Error Type	Specifies the type of error encountered (e.g., syntax error, runtime error, etc.).	ModuleNotFoundError: No module named 'pandas' in GitHub Actions
EH9	Failed solution	This includes explicit or implicit information about any attempted solutions that failed to resolve the issue.	I've trying some things, like this information on this post from January: -LINK- But it didn't work, and one of my partners tried this other post -LINK-.
FI1	Implementation goal	This describes the specific features or goals of the implementation.	So basically what I want is, when I push an annotated tag, I want to first run the test job from build.yml and then once that succeeds, I would like to run the publish job.
OR1	Is it possible to do this?	This includes queries about the feasibility of performing a specific action.	Can I limit force pushes to the GitHub actions user only?
OR2	Search for explanations	This involves looking for explanations or reasons why something is not happening as expected.	What I do not understand is: why does the first Dockerfile build succeeds on Mac/Windows while it fails on GitHub Actions (Linux?).
OR3	Correct logic	Asking if the logic followed is correct. Questioning whether the approach or assumptions and actions are appropriate.	Or setting those as a secret (settings/secrets) should be good enough?
OR4	Best way	Asking what the best or correct path is to take to accomplish something.	What's the best way to test my app using GitHub actions?
OR5	Specific doubt	Asking about a specific question related to GitHub Actions functions, policies, or behaviors.	Do I have to set up SSH keys in my workflow yml file?

RI_id	Relevant Information	Definition	Example
OR6	What do you recommend?	Asking for recommendations for their specific need.	If I wanted to run an arbitrary command and make a PR to the repository, which GitHub Actions should I be looking at instead of reinventing my own Actions?
LE1	Learning specific functions	Asking how to perform specific actions using GitHub Actions.	How can I add a secret in GitHub Actions using my terminal?
LE2	Documentation search	Searching for documentation or examples related to specific GitHub Actions features or commands.	I've landed on Kiwi TCMS as an option for managing our testing system, I had had a brief look and like the look of it - what I'd like to track down before I go much deeper is to see if anyone has experience with how well it works with github actions?
II1	Not working as expected	This describes situations where the developer's implementation does not meet their requirements or functions as expected.	I can build it, but I can't download the artifact.
IN1	Work locally, fail in GHA	This indicates that the code works correctly in the local environment but fails when executed in GitHub Actions.	Everything is working locally on Ubuntu 22.04 LTS and building a Docker image, but not when using the same OS on GitHub Actions.
MI1	Change CI/CD platform	This involves migrating the CI/CD code to GitHub Actions.	What's the equivalent of GitLab's 'artifacts:reports:dotenv' keyword in GitHub Actions?
AS1	Looking for alternatives	Seeking for alternative approaches to achieve a goal or solve a problem without doing what is already known.	Also, is there another way to fetch all pages in Notion below a specified root page?.

Table 2: Detailed Taxonomy of Relevant Information for GitHub Actions

3 Examples of Classification

Below are examples of how sentences can be classified into the RI categories:

1. **Sentence:** *In the step where I run -CODE- to build and test the app, I'm getting an error that -CODE- is an unknown command, despite being installed via bundler in a previous step.*
   RI Classifications: EH1, EH4
   Justification: This sentence describes where the error occurs (*EH1*), as it explicitly mentions that the error happens during the build and test step. Additionally, it provides context on the actions that preceded the error (*EH4*), indicating that the command was installed in a previous step, but is now being reported as unknown.

2. **Sentence:** *This situation is unsatisfactory, because I cannot even run CI on my own PRs in the main repo without them failing due to unavailable secrets.*
   RI Classification: EH6
   Justification: This sentence addresses the cause of the error (*EH6*), which is the unavailability of secrets that leads to CI failures. The dissatisfaction expressed by the speaker is directly linked to this underlying cause.

3. **Sentence:** *I am trying to compile -LINK- in my project.*
   RI Classification: FI1
   Justification: This sentence describes an implementation goal (*FI1*), where the speaker is attempting to compile a specific component within their project, which is a clear objective they wish to achieve.

4. **Sentence:** *As per project requirement I have to run the selenium project into Linux host machine using github actions for that I was trying to install the Edge browser as the host doesn't contains any browser.*
   RI Classifications: FI1, II1
   Justification: This sentence has two relevant components. The first part describes the implementation goal (*FI1*), which is to run the Selenium project on a Linux host using GitHub Actions. The second part

mentions the insufficiency of the implementation (*II1*), highlighting the issue with the lack of a browser on the host machine.

5. **Sentence:** *How can I access the build log information in terminal without going to github.com?*
   RI Classification: LE1
   Justification: This sentence falls under the category of learning specific functions (*LE1*). The speaker is seeking information on how to perform a specific action within GitHub Actions, specifically accessing build log information through the terminal.

6. **Sentence:** *Does anyone have any examples that you might want to share?*
   RI Classification: LE2
   Justification: The speaker is searching for documentation or examples (*LE2*) related to GitHub Actions. The question implies a request for existing examples that could be used as a reference.

7. **Sentence:** *What's the equivalent of GitLab's -CODE- keyword in GitHub Actions?*
   RI Classification: MI1
   Justification: This sentence is classified under migration (*MI1*) because it involves the transition from one CI/CD platform (GitLab) to another (GitHub Actions). The speaker is seeking equivalent functionality in GitHub Actions.

8. **Sentence:** *Is there a simple way to do this without some additional Workflow to monitor and dispatch things?*
   RI Classification: OR1
   Justification: This sentence questions the feasibility of an action (*OR1*). The speaker is asking if there is a simpler method to achieve their goal without adding extra workflows.

9. **Sentence:** *Why does GitHub actions rest API download artifacts by creating a temporary URL?*
   RI Classification: OR2
   Justification: This sentence involves searching for explanations (*OR2*). The speaker is asking for the reasoning behind a specific behavior of the GitHub Actions REST API, which fits into the category of understanding why something is happening as it is.

10. **Sentence:** *Should I be able to deploy my application such way?*
    RI Classification: OR3
    Justification: The speaker is seeking validation of their logic (*OR3*) in deploying the application. This RI type concerns whether the steps being followed are correct and appropriate for the intended goal.