# GitHub Pull Request Review Report Generation Tool

## Problem Statement

ACS want to be able to monitor how TCS are carrying out PR reviews in a number of GitHub repositories:

* how many PR reviews have occurred in a specified time window?
* who submitted each PR?
* who reviewed each PR?
* how many review comments were made for each PR?
* how many requests for changes were made for each PR?
* how long are PRs open for? If very short then is there sufficient scrutiny?
* how many PRs ended up being merged?
* how many PRs ended up being closed?
* are certain source files seeing a lot of review comments? Could that be an indication of a poorly designed module?
* etc

## Source of Truth

Report information needs to be fetched from the GitHub REST API for pull requests (see <https://docs.github.com/en/rest/pulls?apiVersion=2022-11-28> ) plus any other GitHub APIs necessary for obtaining useful information. Access to these is secured so the program will need to be able to programmatically authenticate with GitHub and do so without leaking any user credentials.

If the solution is developed as a standalone CLI executable then a quick solution to the authentication requirement could be to just ask the user to set an environment variable with the value of a GitHub PAT that has the required permissions to read the REST API.

|  |
| --- |
| Note that the program should only read information from GitHub. There must be no modification of any state on the GitHub server! |

## Outputs

TBD. The following are straw man suggestions…

* CSV file
* display on a custom UI (web site or desktop app) with support for generating CSV

## Type of Executable

TBD. The following are straw man suggestions…

* standalone desktop CLI
* desktop application with UI
* small web application

## Technology Stack

TBD. The following are straw man suggestions…

* Go is a good choice for writing native CLI tooling (lots of CLI framework support)
* Java or Go are both good choices for small web applications

Whatever form it takes, the program needs to be easy to distribute and install. It needs to provide useful help information to the user and there should be documentation written that explains the intended usage including all relevant runtime options.

A diagram of a project

AI-generated content may be incorrect.

## Notes

Sending requests to the GitHub REST APIs from the command line demonstrates that it is possible to supply a GitHub PAT for the bearer token.

For example, this curl request returns a list of all the reviews for a given pull request (where the environment variable GITHUB\_PAT has been set with the value of an authorised GitHub PAT) …

|  |
| --- |
| ➜ curl -L \  -H "Accept: application/vnd.github+json" \  -H "Authorization: Bearer $GITHUB\_PAT" \   -H "X-GitHub-Api-Version: 2022-11-28" \  https://api.github.com/repos/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pulls/5/reviews |

The above request will return an array of JSON object data as shown below. Each separate review has its own unique id value…

|  |
| --- |
| [  {  "id": 2913424654,  "node\_id": "PRR\_kwDOOswGsc6tp1UO",  "user": {  "login": "gharley-acs",  "id": 210875831,  "node\_id": "U\_kgDODJG1tw",  "avatar\_url": "https://avatars.githubusercontent.com/u/210875831?u=3ae1a2ce41928eb448c342bb9fb7acaf6e57f9ee&v=4",  "gravatar\_id": "",  "url": "https://api.github.com/users/gharley-acs",  "html\_url": "https://github.com/gharley-acs",  "followers\_url": "https://api.github.com/users/gharley-acs/followers",  "following\_url": "https://api.github.com/users/gharley-acs/following{/other\_user}",  "gists\_url": "https://api.github.com/users/gharley-acs/gists{/gist\_id}",  "starred\_url": "https://api.github.com/users/gharley-acs/starred{/owner}{/repo}",  "subscriptions\_url": "https://api.github.com/users/gharley-acs/subscriptions",  "organizations\_url": "https://api.github.com/users/gharley-acs/orgs",  "repos\_url": "https://api.github.com/users/gharley-acs/repos",  "events\_url": "https://api.github.com/users/gharley-acs/events{/privacy}",  "received\_events\_url": "https://api.github.com/users/gharley-acs/received\_events",  "type": "User",  "user\_view\_type": "public",  "site\_admin": false  },  "body": "Here's the first part of the review @Pkumar2739 with some comments for you to consider. Will provide further feedback soon.",  "state": "CHANGES\_REQUESTED",  "html\_url": "https://github.com/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pull/5#pullrequestreview-2913424654",  "pull\_request\_url": "https://api.github.com/repos/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pulls/5",  "author\_association": "MEMBER",  "\_links": {  "html": {  "href": "https://github.com/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pull/5#pullrequestreview-2913424654"  },  "pull\_request": {  "href": "https://api.github.com/repos/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pulls/5"  }  },  "submitted\_at": "2025-06-11T15:15:32Z",  "commit\_id": "9cdab3c9be00cc23e0af4f167a577576439ff2d0"  },   . . . |

The following request fetches an array of JSON object data describing all the comments associated with the first review in the previous response (id = 2913424654) …

|  |
| --- |
| ➜ curl -L \  -H "Accept: application/vnd.github+json" \  -H "Authorization: Bearer $GITHUB\_PAT" \  -H "X-GitHub-Api-Version: 2022-11-28" \  https://api.github.com/repos/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pulls/5/reviews/2913424654/comments |

The response data includes the following key elements…

* GitHub user who made the comment
* path of the source file commented on
* location in the source file that the comment refers to
* the text of the comment

|  |
| --- |
| [  {  "id": 2137812777,  "node\_id": "PRRC\_kwDOOswGsc5\_bG8p",  "url": "https://api.github.com/repos/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pulls/comments/2137812777",  "pull\_request\_review\_id": 2913424654,  "diff\_hunk": "@@ -121,6 +177,7 @@ public APIGatewayProxyResponseEvent handleRequest(APIGatewayProxyRequestEvent ev\n LogService.logInfo(ionId,\"Received Query Parameter\",\"\");\n String expectedQueryParam=ssmConfig.get(AwsConstants.Expected\_Query\_Param);\n String queryParam = null;\n+ //If query param not null in event then we will get Query Param Value from event Otherwise give error message",  "path": "src/main/java/org/acs/ion/sync/IonSyncHandler.java",  "position": 195,  "original\_position": 155,  "commit\_id": "9f0861a48f474e2f789804dc4b24180168f5b6ef",  "user": {  "login": "gharley-acs",  "id": 210875831,  "node\_id": "U\_kgDODJG1tw",  "avatar\_url": "https://avatars.githubusercontent.com/u/210875831?v=4",  "gravatar\_id": "",  "url": "https://api.github.com/users/gharley-acs",  "html\_url": "https://github.com/gharley-acs",  "followers\_url": "https://api.github.com/users/gharley-acs/followers",  "following\_url": "https://api.github.com/users/gharley-acs/following{/other\_user}",  "gists\_url": "https://api.github.com/users/gharley-acs/gists{/gist\_id}",  "starred\_url": "https://api.github.com/users/gharley-acs/starred{/owner}{/repo}",  "subscriptions\_url": "https://api.github.com/users/gharley-acs/subscriptions",  "organizations\_url": "https://api.github.com/users/gharley-acs/orgs",  "repos\_url": "https://api.github.com/users/gharley-acs/repos",  "events\_url": "https://api.github.com/users/gharley-acs/events{/privacy}",  "received\_events\_url": "https://api.github.com/users/gharley-acs/received\_events",  "type": "User",  "user\_view\_type": "public",  "site\_admin": false  },  "body": "1. The structure of the response body is wrong; the identity of the missing query parameter should be included in the JSON string but it is not.\r\n2. The content of the response body is wrong; it should include the name of the missing query parameter (i.e. the value of `expectedQueryParam`) not null.\r\n\r\nThe above problems are not being caught because the unit test of this behaviour is currently not verifying either of these important characteristics. Please could you address this as otherwise we will be returning a badly-formed error message that does not mention the essential information about the identity of the missing query parameter.",  "created\_at": "2025-06-10T12:46:27Z",  "updated\_at": "2025-06-11T15:15:32Z",  "html\_url": "https://github.com/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pull/5#discussion\_r2137812777",  "pull\_request\_url": "https://api.github.com/repos/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pulls/5",  "author\_association": "MEMBER",  "\_links": {  "self": {  "href": "https://api.github.com/repos/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pulls/comments/2137812777"  },  "html": {  "href": "https://github.com/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pull/5#discussion\_r2137812777"  },  "pull\_request": {  "href": "https://api.github.com/repos/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pulls/5"  }  },  "original\_commit\_id": "bb688a45cca4d1695b781c90b2bc82c98a4bc50c",  "reactions": {  "url": "https://api.github.com/repos/ACSPubsEDSG/fto-iotech-ion-sync-requesthandler-service/pulls/comments/2137812777/reactions",  "total\_count": 0,  "+1": 0,  "-1": 0,  "laugh": 0,  "hooray": 0,  "confused": 0,  "heart": 0,  "rocket": 0,  "eyes": 0  }  },  . . .  }  ] |