# Contributing to this repository

Use this repository to:

- Report bugs in the documentation.

- Propose minor updates to content such as typo fixes or clarifications.

If you have a specific question or issues with the API, please [let us know by contacing GitHub Support](https://github.com/contact).

## API Reference Style Guide

The API reference refers to most of the content under <https://developer.github.com/v3/>. The format of these pages is consistent:

- Endpoints are introduced with an `h2` that describes the API.

For example: `## List issues for a repository`

- The endpoint itself is wrapped in a codeblock. Capitalize the HTTP verb followed by the endpoint URL.

For example: `GET /repos/:owner/:repo/issues`

- Introduce the parameters with an `h3`.

For example: `### Parameters`.

- Write a table with three columns that describes any parameters. The three column headings are:

- `Name`, which identifies the name of the parameter.

- `Type`, which identifies the type of the parameter.

- `Description`, which describes the parameter. Start the description with the phrase \*\*Required.\*\* if it's required. If it's an optional parameter, end the description by listing the default value, if any.

- Provide the endpoint's response. Responses are stored in the \_lib/responses\_ folder.

Optionally, you may choose to include an example. Examples should be introduced with an `h3`, and should occur after the parameters are introduced and before the response.

## Platform Guides

Platform Guides refer to the content under <https://developer.github.com/guides/>. They are longer form content that solve a specific problem for the reader. Each guide follows a three-section pattern:

- An introduction stating any minimum requirements, such as installed dependencies, as well as a description of the problem to solve.

- A body that breaks down the solution to the problem with clear guidelines. Include code samples that are preceded with how the sample could be used.

- A conclusion that summarizes that guide and offers next steps for any advanced topics.

Please submit the full sample code for a guide to <https://github.com/github/platform-samples>.

## Versioning content

Our documentation is single sourced and versioned to also apply to GitHub Enterprise users. We use [Liquid tags](https://help.shopify.com/themes/liquid/basics#tags) to include or exclude content for various builds.

Each GitHub Enterprise release represent a "point in time" for the GitHub.com design and feature set. As a result, the APIs available on GitHub Enterprise are also the same APIs available on GitHub.com at the point when the new version was created. For example, if GitHub Enterprise 2.5 was released on December 15th, the APIs available to users are the same as whatever was available on December 15th. 2.5.x patch releases might introduce bug fixes and updates, but will rarely, if ever, contain brand new API endpoints.

When writing API documentation, we filter sections meant just for GitHub.com using Liquid tags, like this:

```

{% if page.version == 'dotcom' %}

You need to use an HTTP client which supports...

{% endif %}

```

Similarly, content for GitHub Enterprise is versioned based on the release number, like this:

```

{% if page.version != 'dotcom' and page.version >= 2.2 %}

If your GitHub Enterprise appliance has LDAP Sync enabled...

{% endif %}

```