# Contributing Guidelines

Interested in contributing to the AWS Serverless Application Model (AWS SAM)?

Awesome! Read this document to understand how to report issues, contribute

features and participate in the development process. We want to create a

transparent and open process for evolving AWS SAM.

Please read through this document before submitting any issues or pull requests to ensure we have all the necessary information to effectively respond to your bug report or contribution.

## Integrating AWS SAM into your tool

We encourage you to modify SAM to integrate it with other frameworks and deployment providers from the community for building serverless applications. If you're building a new tool that will use AWS SAM, let us know how we can help!

## Submitting feature requests

We love features that help developers build serverless applications faster and further simplify development of serverless applications. If you have ideas for new event sources, new resource types, and new parameters within SAM, follow these steps to submit your feature request:

1. Scan through the list

of

[feature requests](https://github.com/awslabs/serverless-application-model/labels/type/feature).

If you find your feature there, mark it +1. This is a good indication of

interest from the community.

2. If you don't find your feature in our backlog, create a new issue and assign

the `Feature Request` label to it.

Keep in mind that features should be driven by real use cases. Answer the

following questions in the issue that you create:

- Does this feature simplify creating and deploying some aspect of a serverless application?

- Is this feature solving a problem faced by other serverless application developers?

- How do developers work around this problem today?

- How is the proposed feature better than the work around?

## Adding features to AWS SAM

We welcome pull requests to add new features to AWS SAM. Take a look at the

backlog of [Feature Requests](https://github.com/awslabs/serverless-application-specification/labels/feature-request) and pick an item that you find interesting. If the requirements have been well-scoped, feel free to make the change and send a pull request.

If you don't find an item tracking what you have in mind, start a conversation with your plan for implementing

the feature. When defining your feature, keep in mind that one of the core

tenets of AWS SAM is to keep it easy to use while allowing customers access to

use more advanced components, should they so choose.

This repository contains the SAM specification, the code that translates SAM templates into AWS CloudFormation stacks, general information about the model, and examples of common applications. Make enhancements to all of SAM: if you make a change to the specification, please also make the corresponding change to the implementation.

Here are some questions that you should answer in your plan:

- \*\*What is the problem you are solving?\*\*

Example: Creating API endpoints require 24 steps, most of which are boilerplate and repetitive.

- \*\*Describe persona of someone who is facing this problem? This will give you

an understanding of how big of a problem it actually is.\*\*

Example: John just finished coding bootcamp and wants to create a Serverless app. He has

basic coding skills and will be comfortable understanding infrastructure

concepts, but probably be intimidated by the 24 steps required to create an

API endpoint.

- \*\*How do developers work around this problem today?\*\*

Example: Manually click through every step on the website while referring to

"How To" resources on the internet.

- \*\*Describe your proposed solution?\*\*

Example: We are creating a new AWS SAM resource called "API". Here is how it

will look:

```yaml

Type: 'AWS::Serverless::Api'

Properties:

# Name of API endpoint

Name: <string>

# Path to their endpoint. Example: /hello

Path: <string>

# HTTP Method for their endpoint. Example: GET, POST etc

Method: <string>

```

- \*\*How is the proposed feature better than what the work around?\*\*

Example: Developers can write a simple AWS SAM resource which will handle

generating the boilerplate for them.

## Reporting Bugs/Feature Requests

We welcome you to use the GitHub issue tracker to report bugs or suggest features.

When filing an issue, please check [existing open](https://github.com/awslabs/serverless-application-model/issues), or [recently closed](https://github.com/awslabs/serverless-application-model/issues?utf8=%E2%9C%93&q=is%3Aissue%20is%3Aclosed%20), issues to make sure somebody else hasn't already

reported the issue. Please try to include as much information as you can. Details like these are incredibly useful:

- A reproducible test case or series of steps

- The version of our code being used

- Any modifications you've made relevant to the bug

- Anything unusual about your environment or deployment

## Contributing via Pull Requests

Contributions via pull requests are much appreciated. Before sending us a pull request, please ensure that:

1. You are working against the latest source on the \*develop\* branch.

2. You check existing open, and recently merged, pull requests to make sure someone else hasn't addressed the problem already.

3. You open an issue to discuss any significant work - we would hate for your time to be wasted.

4. You propose complete changes - if you make a change to the specification, please also make the corresponding change to the implementation.

To send us a pull request, please:

1. Fork the repository.

2. Modify the source; please focus on the specific change you are contributing. If you also reformat all the code, it will be hard for us to focus on your change.

3. Ensure local tests pass.

4. Commit to your fork using clear commit messages.

5. Send us a pull request, answering any default questions in the pull request interface.

6. Pay attention to any automated CI failures reported in the pull request, and stay involved in the conversation.

GitHub provides additional document on [forking a repository](https://help.github.com/articles/fork-a-repo/) and [creating a pull request](https://help.github.com/articles/creating-a-pull-request/).

## Finding contributions to work on

Looking at the existing issues is a great way to find something to contribute on. As our projects, by default, use the default GitHub issue labels ((enhancement/bug/duplicate/help wanted/invalid/question/wontfix), looking at any ['help wanted'](https://github.com/awslabs/PRIVATE-aws-sam-development/labels/help%20wanted) issues is a great place to start.

## Code of Conduct

This project has adopted the [Amazon Open Source Code of Conduct](https://aws.github.io/code-of-conduct). For more information see the [Code of Conduct FAQ](https://aws.github.io/code-of-conduct-faq) or contact opensource-codeofconduct@amazon.com with any additional questions or comments.

## Security issue notifications

If you discover a potential security issue in this project we ask that you notify AWS/Amazon Security via our [vulnerability reporting page](http://aws.amazon.com/security/vulnerability-reporting/). Please do \*\*not\*\* create a public github issue.

## Licensing

See the [LICENSE](https://github.com/awslabs/PRIVATE-aws-sam-development/blob/master/LICENSE) file for our project's licensing. We will ask you to confirm the licensing of your contribution.

We may ask you to sign a [Contributor License Agreement (CLA)](http://en.wikipedia.org/wiki/Contributor\_License\_Agreement) for larger changes.