![](http://cdn.pocketmine.net/img/PocketMine-MP-h.png)

# PocketMine-MP Contribution Guidelines

You must follow these guidelines if you wish to contribute to the PocketMine-MP code base, or participate in issue tracking.

## I have a question

\* For questions, please refer to the \_#pocketmine\_ or \_#mcpedevs\_ IRC channel on Freenode. There is a [WebIRC](http://webchat.freenode.net?channels=pocketmine,mcpedevs&uio=d4) if you want.

\* You can ask directly to \_[@PocketMine](https://twitter.com/PocketMine)\_ in Twitter, but don't expect an immediate reply.

\* You may use our [Forum](http://forums.pocketmine.net) to ask questions.

\* We do not accept questions or support requests in our issue tracker.

## Creating an Issue

- First, use the [Issue Search](https://github.com/PocketMine/PocketMine-MP/search?ref=cmdform&type=Issues) to check if anyone has reported it.

- If your issue is related to a plugin, you must contact their original author instead of reporting it here.

- If your issue is related to a PocketMine official plugin, or our Android application, you must create an issue on that specific repository.

- \*\*Support requests are not bugs.\*\* Issues such as "How do I do this" are not bugs and are closed as soon as a collaborator spots it. They are referred to our Forum to seek assistance.

- \*\*No generic titles\*\* such as "Question", "Help", "Crash Report" etc. If an issue has a generic title they will either be closed on the spot, or a collaborator will edit it to describe the actual symptom.

- Information must be provided in the issue body, not in the title. No tags are allowed in the title, and do not change the title if the issue has been solved.

- Similarly, no generic issue reports. It is the issue submitter's responsibility to provide us an issue that is \*\*trackable, debuggable, reproducible, reported professionally and is an actual bug\*\*. If you do not provide us with a summary or instructions on how to reproduce the issue, it is a support request until the actual bug has been found and therefore the issue is closed.

- To express appreciation, objection, confusion or other supported reactions on pull requests, issues or comments on them, use GitHub [reactions](https://github.com/blog/2119-add-reactions-to-pull-requests-issues-and-comments) rather than posting an individual comment with an emoji only. This helps keeping the issue/pull rqeuest conversation clean and readable.

## Contributing Code

\* Use the [Pull Request](https://github.com/PocketMine/PocketMine-MP/pull/new) system, your request will be checked and discussed.

\* \_\_Create a single branch for that pull request\_\_

\* Code using the syntax as in PocketMine-MP. See below for an example.

\* The code must be clear and written in English, comments included.

\* Use descriptive commit titles

\* \_\_No merge commits are allowed, or multiple features per pull request\_\_

\*\*Thanks for contributing to PocketMine-MP!\*\*

### Code Syntax

It is mainly [PSR-2](https://github.com/php-fig/fig-standards/blob/master/accepted/PSR-1-basic-coding-standard.md) with a few exceptions.

\* Opening braces MUST go on the same line, and MUST NOT have spaces before.

\* `else if` MUST be written as `elseif`. \_(It is in PSR-2, but using a SHOULD)\_

\* Control structure keywords or opening braces MUST NOT have one space before or after them.

\* Code MUST use tabs for indenting.

\* Long arrays MAY be split across multiple lines, where each subsequent line is indented once.

\* Files MUST use only the `<?php` tag.

\* Files MUST NOT have an ending `?>` tag.

\* Code MUST use namespaces.

\* Strings SHOULD use the double quote `"` except when the single quote is required.

```php

<?php

namespace pocketmine\example;

class ExampleClass{

const EXAMPLE\_CLASS\_CONSTANT = 1;

public $examplePublicVariable = "defaultValue";

private $examplePrivateVariable;

public function \_\_construct($firstArgument, &$secondArgument = null){

if($firstArgument === "exampleValue"){ //Remember to use === instead == when possible

//do things

}elseif($firstArgument === "otherValue"){

$secondArgument = function(){

return [

0 => "value1",

1 => "value2",

2 => "value3",

3 => "value4",

4 => "value5",

5 => "value6",

];

}

}

}

}

```

### RFC and Voting

\* These are big Pull Requests or contributions that change important behavior.

\* RFCs will be tagged with the \*PR: RFC\* label

\* A vote will be held once the RFC is ready. All users can vote commenting on the Pull Request

\* Comments MUST use "Yes" or "No" on the FIRST sentence to signify the vote, except when they don't want it to be counted.

\* If your comment is a voting comment, specify the reason of your vote or it won't be counted.

\* After voting has been closed, no further votes will be counted.

\* An RFC will be rejected if less than 50% + 1 (simple majority) has voted Yes.

\* If the RFC is approved, Team Members have the final word on its implementation or rejection.

\* RFCs with complex voting options will specify the vote percentage or other details.

## Bug Tracking for Collaborators

### Labels

To provide a concise bug tracking environment, prevent the issue tracker from over flowing and to keep support requests out of the bug tracker, PocketMine-MP uses a label scheme a bit different from the default GitHub Issues labels.

PocketMine-MP uses GitHub Issues Labels. There are a total of 12 labels.

Note: For future reference, labels must not be longer than 15 letters.

#### Categories

Category labels are prefixed by `C:`. Multiple category labels may be applied to a single issue(but try to keep this to a minimum and do not overuse category labels).

- C: Core - This label is applied when the bug results in a fatal crash, or is related to neither Gameplay nor Plugin API.

- C: Gameplay - This label is applied when the bug effects the gameplay.

- C: API - This label is applied when the bug effects the Plugin API.

#### Pull Requests

Pull Requests are prefixed by `PR:`. Only one label may be applied for a Pull Request.

- PR: Bug Fix - This label is applied when the Pull Request fixes a bug.

- PR: Contribution - This label is applied when the Pull Request contributes code to PocketMine-MP such as a new feature or an improvement.

- PR: RFC - Request for Comments

#### Status

Status labels show the status of the issue. Multiple status labels may be applied.

- Reproduced - This label is applied when the bug has been reproduced, or multiple people are reporting the same issue and symptoms in which case it is automatically assumed that the bug has been reproduced in different environments.

- Debugged - This label is applied when the cause of the bug has been found.

- Priority - This label is applied when the bug is easy to fix, or if the scale of the bug is global.

- Won't Fix - This label is applied if the bug has been decided not be fixed for some reason. e.g. when the bug benefits gameplay. \*This label may only be applied to a closed issue.\*

#### Miscellaneous

Miscellaneous labels are labels that show status not related to debugging that bug. The To-Do label and the Mojang label may not be applied to a single issue at the same time.

- To-Do - This label is applied when the issue is not a bug, but a feature request or a list of features to be implemented that count towards a milestone.

- Mojang - This label is applied when the issue is suspected of being caused by the Minecraft: Pocket Edition client, but has not been confirmed.

- Invalid - This label is applied when the issue is reporting a false bug that works as intended, a support request, etc. \*This label may only be applied to a closed issue.\*

### Closing Issues

To keep the bug tracker clear of non-related issues and to prevent it from overflowing, \*\*issues must be closed as soon as possible\*\* (This may sound unethical, but it is MUCH better than having the BUG TRACKER filled with SUPPORT REQUESTS and "I NEED HELP").

If an issue does not conform to the "Creating an Issue" guidelines above, the issue should be closed.

### Milestones

PocketMine-MP uses GitHub Milestones to set a goal for a new release. A milestone is set on the following occasions.

- A new Beta release

- A new Stable release

A milestone must use the following format:

```

Alpha\_<version\_number> [release\_title][release\_version]

```

For example:

```

Alpha\_1.4 beta2

```