# Contributing Code

PXT accepts bug fix pull requests. For a bug fix PR to be accepted, it must first have a tracking issue that has been marked approved. Your PR should link to the bug you are fixing. If you've submitted a PR for a bug, please post a comment in the bug to avoid duplication of effort.

PXT also accepts new feature pull requests. For a feature-level PR to be accepted, it first needs to have design discussion. Design discussion can take one of two forms a) a feature request in the issue tracker that has been marked as approved or b) the PR must be accompanied by a full design spec and this spec is later approved in the open design discussion. Features are evaluated against their complexity, impact on other features, roadmap alignment, and maintainability.

These two blogs posts on contributing code to open source projects are a good reference: [Open Source Contribution Etiquette](http://tirania.org/blog/archive/2010/Dec-31.html) by Miguel de Icaza and [Don't "Push" Your Pull Requests](https://www.igvita.com/2011/12/19/dont-push-your-pull-requests/) by Ilya Grigorik.

## Security

If you believe you have found a security issue in PXT, please share it with us privately following the guidance at the Microsoft [Security TechCenter](https://technet.microsoft.com/en-us/security/ff852094). Reporting it via this channel helps minimize risk to projects built with PXT.

## Legal

You will need to complete a Contributor License Agreement (CLA) before your pull request can be accepted. This agreement testifies that you are granting us permission to use the source code you are submitting, and that this work is being submitted under appropriate license that we can use it.

You can complete the CLA by going through the steps at https://cla.microsoft.com. Once we have received the signed CLA, we'll review the request. You will only need to do this once.

## Housekeeping

Your pull request should:

\* Include a description of what your change intends to do

\* Be a child commit of a reasonably recent commit in the master branch

\* Pass all unit tests

\* Have a clear commit message

\* Include adequate tests

## Issue tracker guidance

Going ahead all user facing issues should go to issue tracker in a specific target. For example any issue which impacts microbit user should go to pxt-microbit issue tracker. PXT releases one target every friday and we can look at that target repo issue tracker and triage all issues which are relevant just for that target.

It is possible, issue is underlying PXT issue and can prop up in multiple editors. Even then file it on the target repo where you reproduce it. Scanning PXT repo every Friday is not a manageable thing to do if we file all the issues there. Duplicates will be closed when we continuously release each target.

When to use PXT repo issue tracker?

\* Future enhancement request which is not specific to any targets. Such as delete option for projects in the home screen or a teacher mode.

\* Any bugs by open source developers who are using PXT to build their own editor and running into issues.

\* Some targets don’t have issue tracker which is open such as Minecraft. Guidance for external users is to file in pxt. Internal bugs should still go to pxt-minecraft.

\* Catch all other bugs.