Contributing to sabre projects

==============================

Want to contribute to sabre/dav? Here are some guidelines to ensure your patch

gets accepted.

Building a new feature? Contact us first

----------------------------------------

We may not want to accept every feature that comes our way. Sometimes

features are out of scope for our projects.

We don't want to waste your time, so by having a quick chat with us first,

you may find out quickly if the feature makes sense to us, and we can give

some tips on how to best build the feature.

If we don't accept the feature, it could be for a number of reasons. For

instance, we've rejected features in the past because we felt uncomfortable

assuming responsibility for maintaining the feature.

In those cases, it's often possible to keep the feature separate from the

sabre projects. sabre/dav for instance has a plugin system, and there's no

reason the feature can't live in a project you own.

In that case, definitely let us know about your plugin as well, so we can

feature it on [sabre.io][4].

We are often on [IRC][5], in the #sabredav channel on freenode. If there's

no one there, post a message on the [mailing list][6].

Coding standards

----------------

sabre projects follow:

1. [PSR-1][1]

2. [PSR-4][2]

sabre projects don't follow [PSR-2][3].

In addition to that, here's a list of basic rules:

1. PHP 5.4 array syntax must be used every where. This means you use `[` and

`]` instead of `array(` and `)`.

2. Use PHP namespaces everywhere.

3. Use 4 spaces for indentation.

4. Try to keep your lines under 80 characters. This is not a hard rule, as

there are many places in the source where it felt more sensibile to not

do so. In particular, function declarations are never split over multiple

lines.

5. Opening braces (`{`) are \_always\_ on the same line as the `class`, `if`,

`function`, etc. they belong to.

6. `public` must be omitted from method declarations. It must also be omitted

for static properties.

7. All files should use unix-line endings (`\n`).

8. Files must omit the closing php tag (`?>`).

9. `true`, `false` and `null` are always lower-case.

10. Constants are always upper-case.

11. Any of the rules stated before may be broken where this is the pragmatic

thing to do.

Unit test requirements

----------------------

Any new feature or change requires unittests. We use [PHPUnit][7] for all our

tests.

Adding unittests will greatly increase the likelyhood of us quickly accepting

your pull request. If unittests are not included though for whatever reason,

we'd still \_love\_ your pull request.

We may have to write the tests ourselves, which can increase the time it takes

to accept the patch, but we'd still really like your contribution!

To run the testsuite jump into the directory `cd tests` and trigger `phpunit`.

Make sure you did a `composer install` beforehand.

Release process

---------------

Generally, these are the steps taken to do releases.

1. Update the changelog. Every repo will have a `CHANGELOG.md` file. This file

should have a new version, and contain all the changes since the last

release. I generally run a `git diff` to figure out if I missed any changes.

This file should also have the current date.

2. If there were BC breaks, this usually now means a major version bump.

3. Ensure that `lib/Version.php` or `lib/DAV/Version.php` also matches this

version number.

4. Tag the release (Example `git tag 3.0.1` and push the tag (`git push --tags`).

5. (only for the sabre/dav project), create a zip distribution. Run

`php bin/build.php`.

6. For the relevant project, go to github and click the 'releases' tab. On this

tab I create the release with the relevant version. I also set the

description of the release to the same information of the changelog. In the

case of the `sabre/dav` project I also upload the zip distribution here.

7. Write a blog post on sabre.io. This also automatically updates twitter.

[1]: http://www.php-fig.org/psr/psr-1/

[2]: http://www.php-fig.org/psr/psr-4/

[3]: http://www.php-fig.org/psr/psr-2/

[4]: http://sabre.io/

[5]: irc://freenode.net/#sabredav

[6]: http://groups.google.com/group/sabredav-discuss

[7]: http://phpunit.de/