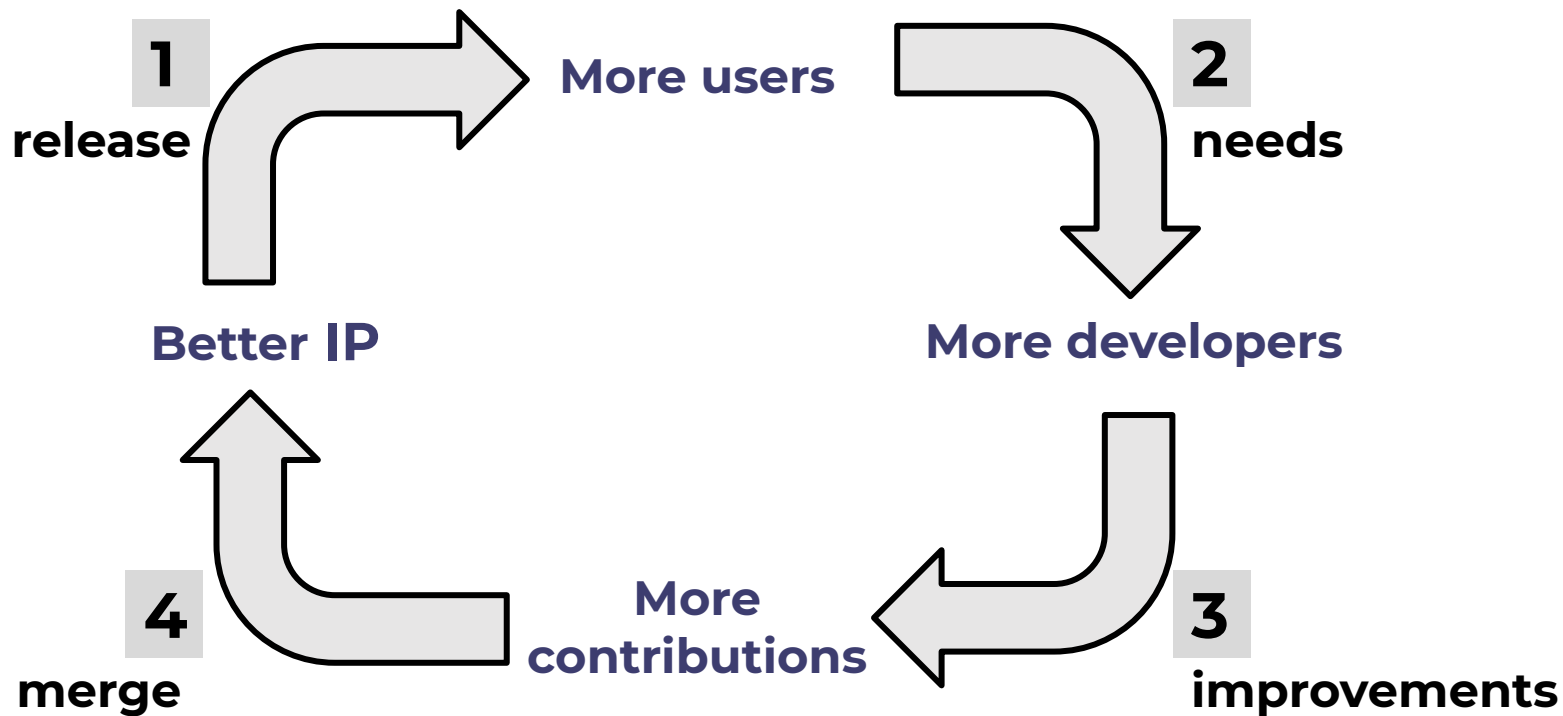


Open Source Project Activities and Maturity

Introduction and guidelines

Continuous improvement to grow an open source project's ecosystem



Terminology

Developers: people who code, core, extensions, modules, themes for themselves or for their customers.

Note: describe here who are they in your community or your project, use personas if possible

Contributors: people who are key active contributors to the open source project, coders, but not only, a lot of contributors actually don't code at all.

Note: describe here who are they in your community or your project, use personas if possible

Corporate contributors: people who are working for a company that is using/integrating the project, and that company wants to improve the project as part of its strategy

Note: list here a few companies that are contributing, if any already

Intellectual Property: code, documentation, artwork, trademarks, domain names, etc.

Note: list here brands, logos, ...

Status icons



Green check mark: the status of the project is ok regarding the section and your objectives



Blue eye mark: you need to investigate because you miss information, or the definition of the objective for this section is not clear



Orange exclamation mark: important problems are identified and actions to solve them are listed or in work in progress



Red exclamation mark: blocker, serious issue, or nothing is done yet; or even worse, bad things for the project have been done

Status



Describe the state of the project regarding this section, and then use the right status icon color.

You can update it when there is an evolution.

And don't forget to update the icon on the main dashboard.

Problems & objectives

What needs to be solved or improved, and eventually why.

Keep it short, and add long sources and references (links to web page, document, ...) in the notes of the slides

Actions

The aim is to be able to create a backlog with all the action of all section. You can then add a link to the tickets you created in your project solution (GitHub issues, Jira, Tuleap, Bugzilla, ...) with more details.

Then, it will be possible to sort them by priorities, difficulty, and start the work with the different teams. And of course, to follow the implementation.



Dashboard

Strategic objectives

Name of your project:

- Objective 1
 - Objective 2
 - Objective 3
 - Objective 4
-

Name of your project Open Source Project

Last update - November 2019



Encourage

USERS



Community development

Project license

User Documentation

Discussion Platforms

Software construction maturity

Project executables published

Project Install Automated

Project Bug Tracking

IP management activities

Project license documented

Repositories protected

Dependencies documented

Committer Governance

Contributions audited

Provenance tracking

Provenance Management

Trademark management

Committers indemnified

Encourage

DEVELOPERS



Mission Statement

Contribution guidelines

Developers Platform

Complete source published

Project build documented

Project tests documented

Encourage

CONTRIBUTORS



Code of conduct

Governance

Events & promotion

Project build automated

Project tests automated

Basic architecture description



Encourage

CORPORATE CONTRIBUTIONS



Community Development

Status



Problems & Objectives

Actions



Status



Problems & objectives

Actions



Discussion platforms

Forums, Q&A, chat, mailing lists, news groups, social networks... any service where users discuss and support each others

Status



Problems & objectives

Action



Mission statement

Why the project has been started, what is it supposed to solve?
Please note that this about the project itself, not about an eventual company supporting it.

Status



Problems & objectives

Actions



Contribution guidelines

All the useful information to contribute to the project in many ways, and not only with code

Status



Problems & objectives

Action



Developers platform

The “forge” where code contributors collaborate, usually a bug tracker, a doc and a chat.

Examples: IRC, Slack, Wiki/Doc, Jira, Tuleap, Bugzilla, GitHub, project kanban,

Status



Problems & objectives

Actions



Code of conduct

Rules to follow when contributing to the project or using the services to collaborate around the project, the discussion are respectful, polite, and based on good will. It must be enforceable.

Status



Problems & objectives

Actions



Governance

Description of how decisions are taken, including the roles in the project (people and companies/organisations), and how to obtain them.

Status



Problems & objectives

Actions



Events & promotion

Dedicated “big” events, business events, meetups, local users groups, blogs, videos and marketing in general...

Status



Problems & objectives

Actions



Software Construction Maturity

**Project executables
published**

The software is distributed, and it is possible for the users to install it and to run it without having to build it.

Status



Problems & objectives

Actions



Project install automated

The software comes with an installer or with a kind of assistant for the configuration, so it's easy to install and to start.

Status



Problems & objectives

Actions



Project bug tracking

The community can report bugs and improvement/feature requests, and follow the work done on the project.

Status



Problems & objectives

Actions



Complete source published

In addition to the executables, the source code is available to download

Status



Problems & objectives

Actions



Project build documented

First step: documentation and scripts provided to the community so anyone can build the project with the sources.

Status



Problems & objectives

Actions



Project tests documented

First step: documentation and scripts provided to the community so anyone can launch the tests against a build

Status



Problems & objectives

Actions



Project build automated

Step 2: builds are automatised every nights with the merged code of the day, or eventually also for each contribution in order to be able to test as soon as possible

Status



Problems & objectives

Actions



Project test automated

Step 2: after each build, tests are launched automatically and a report is built and published for each build

Status



Problems & objectives

Actions



Basic architecture description

Designs and other description of the different components of the software and their interactions, including the core, the dependencies and eventual native extensions/modules

Status



Problems & objectives

Actions



IP Management Activities

Project license documented

From intellectual property's point of view, with the list of license used or approved to be used in the dependencies, written by a lawyer

Status



Problems & objectives

Actions



Repositories protected

Only selected and approved people can modify the code in the repositories of the project

Status



Problems & objectives

Actions



Dependencies documented

The complete list of dependencies, with versions and licenses is available, maintained and controlled.

Status



Problems & objectives

Actions



Committer Governance

Rules for pull request validation and how to become a committer, or a maintainer, and eventually describe the mentoring and other mandatory steps.

Status



Problems & objectives

Actions



Contributions audited

Contributions are merged after a process to check that their quality. The process is described in the developer documentation.

Status



Problems & objectives

Actions



Provenance tracking

Regular audit of existing source code, and eventually automatic check of pull request before merge, to check it is original, to protect users.

Status



Problems & objectives

Actions



Provenance management

Management of source code authors, who they are, and their employer, in order to check that there is no conflict of interest regarding copyright or “droit d’auteur”

Status



Problems & objectives

Actions



Trademark management

The name of the project and its logo are protected trademarks and are part of the IP to manage and protect

Status



Problems & objectives

Actions



Committers indemnified

Legal insurance for people who manage and contribute to the project, to protect them if they are sued by a copyright holder or a patent troll

Status



Problems & objectives

Actions



Credits

Originally designed and built by **Antoine Thomas** for the **PrestaShop** open source project and company, thanks to:

- **Stephen Walli** for his advices and encouragement, and of course, for his list of “open source project communities patterns and practices” (look for his articles and presentations)
- **Aurélien Pelletier** for the design discussions and suggestions.
- **Tristan Lehot** for the look and colours.

And thanks a lot to all the fantastic people who spent time to answer to my questions, to read and to discuss this document. You made it too!

Available under **CC-by-sa 4.0 international**

Original template available at: github.com/PrestaShop/open-source/tree/master/templates