

Always Improving: Making the Contribution to Repos Better



Code.gov

Following

May 7 · 5 min read



The best thing about experience — no matter the project, the platform, or the program — is how you grow from it. There is always room for improvement; and with experience, improvement is sure to follow. At Code.gov, we recently wrapped up a User Experience (UX) project to better understand the challenges users face when contributing to our code repositories. Our goal in conducting this research, which included contributor interviews and analyses of best practices, was to recognize what changes were needed to increase usability and make our repositories more contributor-friendly.

Barriers to Contribution

Four items were regularly identified as barriers to contribution:

1. **Inactive project/community.** One contributor noted “If I see a project hasn’t been modified for a year, I don’t see a point of interacting with it because it’s probably not being used anymore.”
2. **Lack of quality documentation.** According to GitHub’s Open Source Survey, “Documentation is highly valued, frequently overlooked, and a means for establishing inclusive and accessible communities.”
3. **Non-welcoming community.** A welcoming, receptive and responsive community is an important part of open source. Being welcoming means setting positive expectations for behavior, while being responsive means providing some sort of acknowledgment within a couple of days to contributors. Being receptive means being open-minded and clear about the types of contributions that are accepted.
4. **Few external contributions.** Similarly to an inactive project or community, if a project has few external contributions several interviewees noted they would be less likely to contribute themselves as they are unsure whether their contributions would be valued or merged.

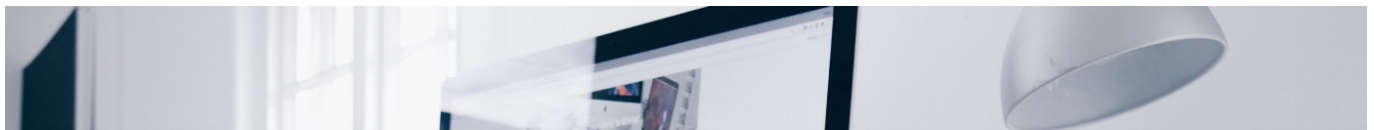




Photo by Christopher Gower on Unsplash

Actionable Recommendations

To address the identified barriers to contribution, we identified actionable recommendations that the Code.gov team will be implementing over the next several months. The recommendations listed below are offered in no particular order or priority.

Inactive project/community

Code.gov needs to review some of our lesser-used repositories to consolidate, if that is possible. We will also need, at the very least, to ensure that a repo's documentation is up-to-date and specifies why certain repositories are not used or updated as often as others.

Lack of quality documentation

In order to establish a more inclusive and accessible community, Code.gov recommends that we:

- Value and prioritize documentation by including explicit language in contributing files
- Develop a “New Contributors” section in the documentation



- Review current issues in repos to see if more information could be added to explain the problem, offer goals, and suggest a fix or outcome, when possible
- Create standard document templates for every Code.gov repo (e.g., issue template, CONTRIBUTING.md, etc.)
- Create base service level guidelines for the community (e.g., expected time for a reply when a new issue or PR is submitted, expected time to first code review, etc.)
- Update project roadmap to reflect current project priorities (in GitHub and on Code.gov)
- Outline the contribution review process
- Identify questions and issues that have been encountered (especially if issues have appeared more than once) to see if these should be included in the README
- Create standard issues tags to use in all repos, be sure these include tags for difficulty level (“first-timer”), and offer estimated time commitments
- Conduct usability testing regularly to surface any additional issues with cloning the repo, standing it up locally, and making a contribution
- Offer sufficient quick start information in the documentation to ensure that potential contributors are able to set up the dev environment successfully
- Ensure documentation is system agnostic
- Focus documentation towards external users who want to jump in and fix things
- Outline the inter-connectedness of the repos (front-end, style guide, component repos, etc.)
- Add integration badges
- Setup automated testing



- Call out kudos to contributors in a public way (e.g., newsletter, meetings, Slack, blog)
- Recognize all contributors in a consistent way
- Communicate our goals and show people their potential impact
- Organize office hours for developer Q&A

Few external contributions

In order to increase the diversity of contributions to Code.gov from external contributors, Code.gov will:

- Encourage external contribution explicitly in documentation and implicitly in communication wherever the community congregates
- Look into participating in Google Summer of Code for 2020 (and other events like All Things Open, Code for America, etc.) to encourage external contribution
- Gather feedback from participants/contributors via survey after events to identify and resolve any pain points

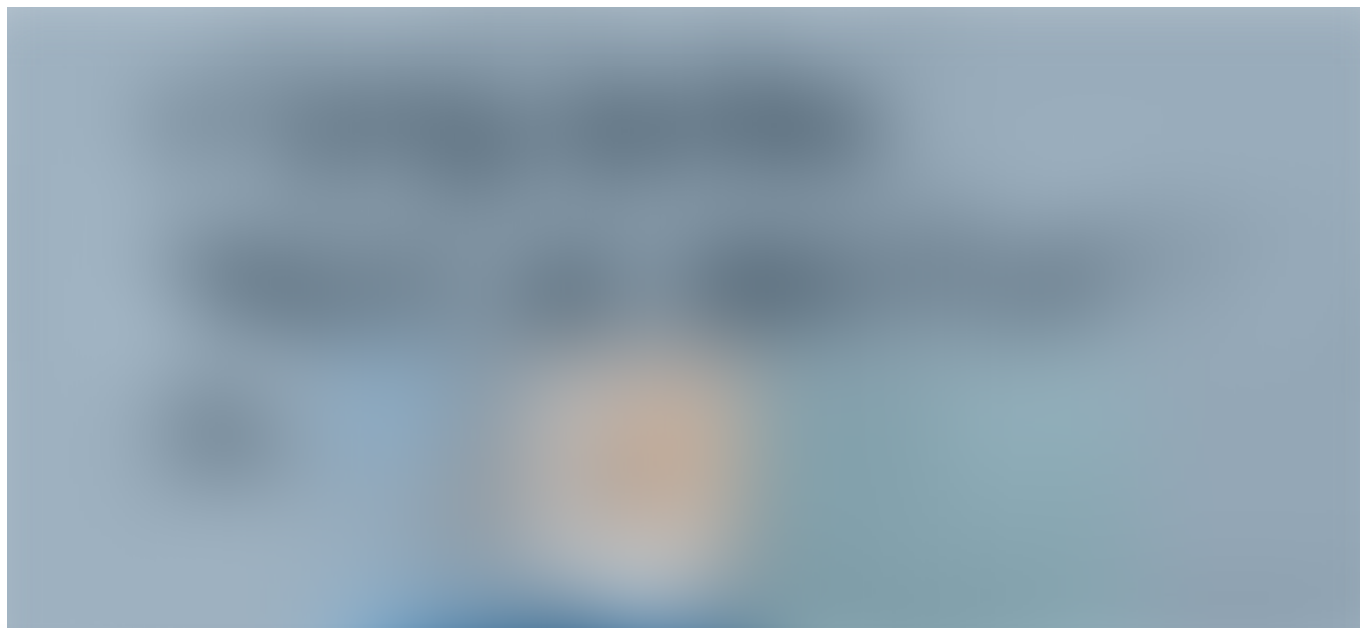


Photo by Kobu Agency on Unsplash

Technology is always in a state of flux, but here at Code.gov, we are always working to improve what we do and how to make our platform a better experience for you. Through our website, we offer you an opportunity to work with open source software (OSS) that belongs to you. This is code that has been bought and paid for by you, one of many reasons why we call the projects at Code.gov “America’s Code.” We wish to inspire you in creativity and ingenuity, and so we work to make the process of contributing easy. Code.gov offers you a chance to fulfill a civic duty on a digital platform, one line of code at a time.

We want to hear your story concerning your experience with Code.gov and how we have inspired you, so share your stories with us through email, Twitter, LinkedIn, or the Comments section below. We hope to meet you online, to answer whatever questions you may have for us; and we look forward to your contributions.

Code on.

Special thanks to Code.gov’s UX designer, Krissy Kimura, whose research made this article possible.

[Open Source](#)[Civitech](#)[Civic Engagement](#)[Civic Innovation](#)[Civic Hacking](#)

9 claps



WRITTEN BY

Code.gov





CodeDotGov

The primary platform for sharing America's code.

Following

See responses (1)

More From Medium

Related reads



Geographical Data Visualization with Mapbox



Todd Birchard in Hackers and Slackers

Dec 11, 2018 · 8 min read ★



369



Related reads

Create categorical choropleth with Python



ZhongTr0n in Towards Data Science

May 22 · 5 min read ★



90



Related reads



Exploring and Visualizing Chicago Transit data using pandas and Bokeh — Part I (intro to pandas)



Colin Patrick Reid in Towards Data Science

May 30, 2018 · 12 min read



309

