Using GitHub to open up your software project

Patricia Herterich • Research Repository Advisor
@pherterich • p.s.herterich@bham.ac.uk
Research software management, sharing and sustainability workshop
University of Birmingham • 09.01.2017
Overview

A short tour through GitHub’s features

How to make your project open
- Plan & Communicate
- Collaborate
- Share openly
- Preserve
- Get credit
Git & GitHub

Git

- Git is the command line software handles version control on your repository.
- When you use GitHub, Git is always doing its thing behind the scenes.

GitHub

- GitHub is a service that hosts your repository online and helps you work with contributors.
- It’s a web interface for version control
- Much of your work can be done in GitHub
GitHub features

- A central repository hosting all the content and changes you (and your collaborators)
- Collaboration options such as
  - Issues to track bugs, tasks, ideas
  - Labels to organise the issues (according to content, priority, level of difficulty, etc)
  - Comments for commits and merges
Using GitHub for open research projects
Communicate your project scope

The more people understand and get excited about what your work is and what it means, the more people will join in and help (or re-use).

Create a readme to show

- what you’re doing, for who, and why
- what makes your project special and exciting
- how to get started
- where to find key resources.
# Open Canvas

**Project:** Title

<table>
<thead>
<tr>
<th>Problem</th>
<th>Solution</th>
<th>Unique Value Proposition</th>
</tr>
</thead>
<tbody>
<tr>
<td>The top 1-3 problems you want to solve</td>
<td>Outline your proposed solution for each problem</td>
<td>A clear message that states what you offer and why you are different. Can be derived from: 1. The main problem you are solving 2. The finished story and benefits users will have by using your product</td>
</tr>
</tbody>
</table>

**Example:** Square - start accepting credit cards today

<table>
<thead>
<tr>
<th>Key Metrics</th>
<th>User Profiles</th>
<th>User Channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>How will you measure success?</td>
<td>Target audience and early adopters</td>
<td>List how you will gain new users</td>
</tr>
<tr>
<td>Who are you building this for? Who will your early adopters be?</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Resources Required</th>
<th>Contributor Profiles</th>
<th>Contributor Channels</th>
</tr>
</thead>
<tbody>
<tr>
<td>What do you need to build an MVP (minimum viable product)? Design, Development, Expertise, Hardware requirements and other costs</td>
<td>Contribution types and ideal contributors</td>
<td>What do your contributors look like? Be sure to include the different expertise you outline in “Resources Required”</td>
</tr>
<tr>
<td>List how you will gain new contributors</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Adapted from the Lean Canvas (https://leanstack.com/lean-canvas/) CC-BY-3.0
Roadmapping

A roadmapping organizes the tasks that need to be done on a project around milestones. This helps potential contributors understand the current status of your project and where it’s going next. A roadmapping can also express your vision for the project.

- Project Summary & Welcome (optional)
- How to get involved (optional)
- Timeline with milestones
  - Completing a feature
  - Short, medium & long term
  - Hard deadlines (dates), working around events
  - Create issues for all tasks and assign them to milestones
Invite collaboration

Create a contributing.md file

- Welcome
- Link to relevant resources
- Testing instructions
- Environment details
- Protocols on submitting changes and reporting bugs
- Style guide/coding conventions
- Code of Conduct
- Recognition model
- Contact information
Leading an open project

- Write good documentation (README, contributing, CoC)
- Respond to Issues in a timely fashion
- Watch your repository
- Build trust by squashing bugs
- Be consistent in your activity
Ownership - the legal stuff

- Employer is first owner of anything produced ‘in the course of employment’
- See UoB Code of Practice for Research, section 5:
  “The University does not in practice assert its ownership of the copyright” (for some limited types of materials) [...]

“However, the University retains its right to use and reproduce such materials for internal educational purposes whilst recognising the author’s moral rights.”

“Unless Intellectual Property is assigned to a research Sponsor or Funder as a condition of grant or contract, intellectual property and any intellectual property rights therein developed during research by Staff in the course of their employment belongs to the University of Birmingham.”

Licensing

As the owner, you have the right to sell and license your work to control how & when it is used and by whom.

http://choosealicense.com/

Which of the following best describes your situation?

I want it simple and permissive.

The MIT License is a permissive license that is short and to the point. It lets people do anything they want with your code as long as they provide attribution back to you and don't hold you liable.

jQuery, .NET Core, and Rails use the MIT License.

I'm concerned about patents.

The Apache License 2.0 is a permissive license similar to the MIT License, but also provides an express grant of patent rights from contributors to users.

Android, Apache, and Swift use the Apache License 2.0.

I care about sharing improvements.

The GNU GPLv3 is a copyleft license that requires anyone who distributes your code or a derivative work to make the source available under the same terms, and also provides an express grant of patent rights from contributors to users.

Bash, GIMP, and Privacy Badger use the GNU GPLv3.
Licensing

choosealicense.com is created by the GitHub people, so you can easily assign a licence when creating a new repository.

If you don’t choose a licence, you retain all rights to your source code and that nobody else may reproduce, distribute, or create derivative works from your work.

BUT: if you publish your source code in a public repository on GitHub, you have accepted the Terms of Service which do allow other GitHub users some rights. Specifically, you allow others to view and fork your repository within the GitHub site.
Preservation

GitHub is awesome, but it might not be around forever & you can delete things any time

- Archive your repository, especially if your work underpins a journal article etc and you refer to your code to allow for the reproducibility of results
- Figshare and zenodo both provide integrations with GitHub that allow you to easily archive your repo and get a Digital Object Identifier (DOI) for it.
Code as a Research Object
Mozilla Science Lab

Get credit for your code!
Archive your GitHub code repository to Figshare and receive a citable DOI.

GitHub
Edit, share and improve your code in a collaborative environment.

Mozilla Science Lab
Tools to get your research on the web.

figshare
Persistent, archival outputs.

Making Your Code Citable
© 10 minute read

Digital Object Identifiers (DOI) are the backbone of the academic reference and metrics system. If you’re a researcher writing software, this guide will show you how to make the work you share on GitHub citable by archiving one of your GitHub repositories and assigning a DOI with the data archiving tool Zenodo.

ProTip: This tutorial is aimed at researchers who want to cite GitHub repositories in academic literature. Provided you’ve already set up a GitHub repository, this tutorial can be completed without installing any special software. If you haven’t yet created a project on GitHub, start first by

https://mozillascience.github.io/code-research-object/

https://guides.github.com/activities/citable-code/
Software is a critical part of modern research... yet there is little support for its acknowledgement and citation.
Software citation

- Currently heavily dependent on software having a DOI and thus following the style of a “classical paper citation”
- Software Citation Tools are currently in development at https://mozillascience.github.io/software-citation-tools/
- Help the team by joining the discussion:
  https://github.com/mozillascience/software-citation-tools/issues
Measure impact

Your code has a DOI? - check for Altmetrics!
Measure impact

Software packages in **pypi** and **CRAN**? - depsy.org
Collaboration is great, but fully open...?

University of Birmingham pilot of GitLab for researchers
<table>
<thead>
<tr>
<th><strong>GitHub</strong></th>
<th><strong>GitLab</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>● Free account means all your repositories are public</td>
<td>● Like GitHub, but run locally</td>
</tr>
<tr>
<td>● Private repositories need a personal subscription $7/month</td>
<td>● All data stored on campus</td>
</tr>
<tr>
<td>● US based company</td>
<td>● Only accessible from the University network</td>
</tr>
<tr>
<td></td>
<td>● Unlimited private repositories – within reason…</td>
</tr>
<tr>
<td></td>
<td>● Unlimited data storage – within reason…</td>
</tr>
</tbody>
</table>


Published under a Creative Commons Attribution 4.0 Licence