Bill Hunt   @krues8dr

Documentation

I'm here today to talk about documentation. I've spent the last few weeks going through the dozens of projects that Sunlight Labs had created over the last ten years, trying to get them all into a condition that people could re-use them – so this has been on my mind a *lot*. Now, documentation is just a boring way of saying…

Bill Hunt   @krues8dr

Write Stuff Down and Share It

… "Write Stuff Down and Share It" And really, it doesn't need to be any more complicated than that. If you're sharing data, or writing software, or hosting an event, or just know how to do something – you should always take the time to write stuff down and share it.

So, what exactly are we talking about when we say "stuff"? Everyone in this room already knows how important open access to data is. And I know many of you are probably already thinking about transparency around policy. Networks - the people who are part of your process. Write them down, give them credit, elevate them. And of course if you're from a research background, you know how important it is to share your process and methods. Here's an example of an open process that you all should be familiar with.

This is my most popular shared documentation - my recipe for vegan potato salad. These are clear, step-by-step instructions that anyone can follow. Anyone can take and use my recipe however they want. They can change it to suit their tastes, adding more salt or removing the dill - though why you'd want to is beyond me, dill is awesome. This right here, this is open data.

Bill Hunt   @krues8dr

Why Is This Hard?

- It's easy to overlook

- Politics

- Fear, Uncertainty, Doubt, Intimidation

- Not having the time

For lots of us who write for a living, or share data, these things aren't always intuitive or easy. Why is this hard? First, it can be easy to forget. The things you do every day are the easiest to overlook. Internal - and external - politics can certainly cause resistance as well. The big four - fear, uncertainty, doubt, intimidation - things that silence people, or cause them to silence themselves. Being worried about your credibility, now or in the future if your opinion changes. And, of course not having - or, making - the time to do it.
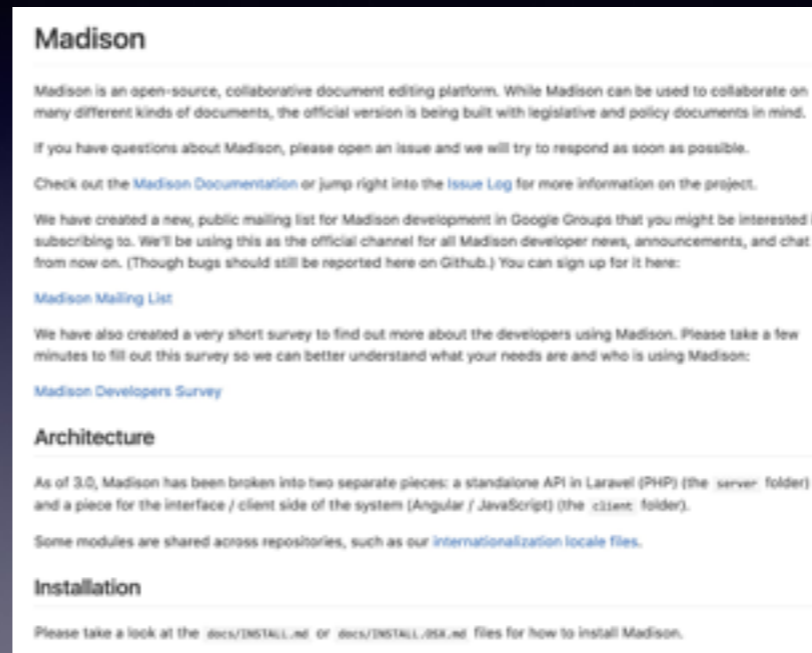
Bill Hunt   @krues8dr

Who?

- **Does this solve my problem?**

- What is this?

- What does it do?

- Who is it for?

- How do I use this?

When you're writing stuff down, the first thing to think about is *who* you're writing this down for. Always think about your audience, where they're coming from. They are looking at your documentation to figure out a simple question: does this solve my problem? Whether you're sharing data, tools, or process, you should let them know : What is this, What does it do, Who is it for, How can they use it?

Here's an example from one of the projects I worked on for years. Madison is a collaborative document platform, intended for legislation and policy, we start there. It's software, but our audience is not just developers, it's also government officials interested in collaborative legislation or policy - and we start to tell them that in the second sentence. Then we tell them how to use the project and how to get in touch with us. This could even be better, though - a feature list and some screenshots are always nice.

After you've answered those questions, think about what else you can share. As I said in my introduction - Open Everything, Please! If you know how to do something, write that down! There are probably lots of things you're doing that you think are mundane or unimportant - but I *promise* someone is probably trying to solve a problem that your knowledge can help with. And the easiest way to share that knowledge is - you guessed it - to write it down and share it!

Bill Hunt @krues8dr

Documentation Driven Development

If you're a developer, I want to very quickly mention documentation driven development here. You've probably heard of test driven development - this is pretty similar. You write a bit of documentation for the piece of your project that you're working on, *then* write your tests based on that documentation, and *then* write your code from the tests. It's a simple step to make sure your documentation stays up to date with your code - which, as we all know, is a big problem.

How?
Licenses

Ok, enough developer talk, let's move on to something far more boring - licensing. Don't worry, I'm not going to go deep on legal stuff here. The short version is, you should tell people *how* they can use your stuff - if you don't tell them explicitly, they won't know.

Bill Hunt @krues8dr

How?

Licenses

As an example, I've done a lot of work in transparency of city and state law. Although the law is legally supposed to be the people's public property, often there are copyright restrictions on portions of the law. A good example is building codes - in many places, you can't get access to those without paying fees. So, even though people can *read it*, they can't *share it* - and that makes it harder to get access. And without access, there's no equity.

Some Licenses I Like

- Creative Commons Zero - puts a work into the public domain.

- GPL v3.0 - if you change it, you have to share your changes.

- MIT - if you change it, you don't have to share your changes.

- **http://choosealicense.com/**

Here are some licenses I like. Creative Commons Zero is a way of putting a document, or photo, or video into the public domain. Anyone can use it or edit it or remix it for anything. GPL 3 is mainly for software, and basically it requires that anyone who uses your code has to share any changes they make to it - it's a sort of "Pay it forward" model. This is the default license we chose for all of the Sunlight Labs projects that we're shelving. There's also the MIT license is a bit different in that anyone can use or edit your code, but they *don't* have to share their changes. This is more permissive, and makes it easier to use your code in other commercial works. There are lots of other licenses, though - choosealicense.com is a great resource for making sense of them in plain language.

Where

- Google Drive/Docs/Sheets

- GitHub

- Medium/Blogs

- Twitter/Facebook/Social Media

- Zines/Posters/Books/Non-digital!

Where should you share your work? Lots of places you're probably already familiar with. Google Drive is a great place to host documents. If your audience is more technical, consider GitHub - I put the HR Manual we wrote at OpenGov Foundation up on GitHub and people have taken it and used it at other organizations! If you have your own blog or one on Medium, that's a great place for long-form thoughts. Of course Twitter, Facebook, and the usual social media sites are great paces for having conversations in the open. And last, going back to my roots a bit here - zines are awesome! Posters can get a message across as well! If you have the time, write a short book! Don't overlook the non-digital space if you really want to engage with people. Write letters!

Bill Hunt   @krues8dr

Sharing
TransparencyCamp

So, to wrap this up, I just want to encourage you all to take some time tonight, tomorrow, or next week to write down what you've learned here at TransparencyCamp, and share it. The new insights that we've all gained here over the last two days can continue to grow and bloom, and others can build on these ideas and create even more new things - but only, *only* if everyone spends the time to document their journey. So please, please make sure to write your stuff down and share it.

And on that note, I also want to point out that we have very well-written documentation on how to host your own Transparency Camp. A lot of amazing people have contributed to putting together Transparency Camp over the years, and I want to give a shout out in particular to Laurenellen McCann and all the other former Sunlighters who put these resources together. We could not have done it without them. And now, you can too - if you liked what we did this year, please feel free to start up your own Transparency Camp! Remember - sharing is caring.