□□ How to work with a mentor – Interview with Common Syllabi

Pierre Depaz and Tobias Schmidt developed Common Syllabi in the 11th round of the Prototype Fund. With Pierre being relatively new to the open-source universe he decided it'd be useful to work with a mentor for the project: Tobias, who has participated in a lot of open-source projects for quite some time. In this interview Pierre talks about the benefits of this model, what the mentorship looked like and gives some advice to people who are also looking for support.

While developing Common Syllabi you worked together with a mentor – why did you decide to do that?

When I started to think about the idea for Common Syllabi, I immediately thought about sustainability for the project: just because you build it, it does not mean that people will use it, and this gets less and less likely throughout the project lifespan. Common Syllabi is a platform to archive and share learning materials across universities. Because of the archival part, there needed to be particular attention paid to the sustainability of the project, both from a technical and organizational perspective.

I decided to work with Tobias as a mentor, because he already had experience in these things, having participated in large and successful open-source projects from the beginning. He could therefore bring some past experience and technical expertise regarding the reliability of the project and its accessibility. In the end, this was a great decision, and helped a lot to kick start the project on solid foundations.

Did you know each other before or did you start looking for a mentor once you knew you wanted to do the project?

Tobias and I knew each other from before. I met him as he was working on an open-source project, the Prometheus monitoring system, at his former company and was very interested in the organizational choices that can make such an open-source project sustainable, as well as in the whole technical process: how do you decide what to work on, how do you design the project from the start, who contributes and how, etc.

Coming from a political science background myself, and having just switched to game development, I knew a little bit about programming, and a little bit about open source, but only as

an amateur/from an outsider perspective. That being said, we had already hacked on a few side projects together, so we knew that we also got along when in a work situation.

To be honest, I would not know where to start if I had to find a mentor once the project had started, and the fact that I knew Tobi already gave me the confidence, that I could deliver what I would set out to during the funding phase.

What were the benefits from this mentorship for you? What would have been different without the mentor?

For me, the benefits were huge. Besides giving me confidence that I was not in this alone, I also learned a lot from a technical perspective: working with an experienced professional is invaluable. This involved coding practices, but also documentation, repository organization, devops, etc.

Tobias: for me, I just really appreciated being able to think about a project at the high level without having to spend time implementing technical details or doing menial work. This mentorship helped me develop more manager skills, as I help junior developers accomplish tasks at the company I am currently at. I also liked the opportunity to write some code in Go!

What did the mentorship look like?

Tobias was on the official application, but we made sure to discuss all of the practical details before we officially applied. This meant a couple of things: as little administration as possible for Tobias (since he already has a full time job, he could only commit a few hours a week), and bi-weekly meetings for checking in on progress and setting out subsequent goals, a little bit in the agile style of working.

The first meetings were focused on the architecture and overall design: I did a lot of the user research, and based on what I found of how people are using current tools, or how they would like to use future tools, we decided on a design that would be lightweight and reliable. The second part of the funding phase was on the same schedule, but rather about code and technical details of implementation. I would work on the design and write down some questions and issues that I encountered, and when we met we would do code reviews to see what worked, or what did not, and how to move forward.

In terms of advice, I would say that there are a few things that were important. First, ideally find someone that you also appreciate at the human level, which makes it a lot easier and fun to work with. Even if you meet someone specifically for this, it would be important to meet at least once before any official planning to see how the mutual feeling is. Second, in-person meetings are a lot faster and more productive than online, when it's about designing or debugging. We only did in-person meetings, which helped to cover a lot more topics and deal with digressions in a more pleasant manner. Third, I would say it's important that there has to be something that the mentor gets out of it as well, whether it's honing some skills, trying a new technology, or even just believing in the project; this helped to keep motivation and focus. Finally, an important part was, that we were both flexible to adapt to each other's schedules. All of this made sure this was remaining a fun and engaging project for both.

Do you have any other tips or ideas for people who want to do a project but are missing a certain skill in order to realize it?

Tips for applicants would be to (1) believe in the idea and (2) scope it down! Even if you don't have all the team members or skills to develop an idea, it's easier to get people on board once you have a clear idea (which the application process helps to formalize), and it's always possible to find some people for specific tasks along the funding phase.

That being said, it's also important to know what are the people/skills that you are missing: it's easy to deal with known unknowns, but more complicated with unknown unknowns (you don't know what you don't know, so it's hard to ask for help about it!). Doing a bit of research on OSS organization (such as browsing the OSAOS Handbook, or OpenCollective) helps a lot in figuring out the main roles in an OSS organization and in identifying where you need help.

Revision #2
Created 3 August 2023 10:15:36 by patricia
Updated 3 August 2023 11:33:39 by patricia