

Transition Document - VP Projects

Due to the nature of the projects, VP Projects is a very cyclical role and thus does not have any major carryover tasks from one year to the next. Since there has not been a lot of structure in the past, I spent the last year trying different things and seeing what works. At the end of this document, I will list some problems and potential improvements that I would have focused on for the upcoming year, but of course, it is up to you to decide.

Here is a month-to-month outline of VP Projects responsibilities:

April

- Developer onboarding

- Unless developers have been on internal team (and graduated to project teams), you will need to onboard new developers with our tech stack and development process. We have also run one in-person onboarding session per semester in the past ([Google Slides link here](#)), but it is up to you whether or not you think this is necessary.

- While developers are onboarding, you can use the **#onboarding-developers** channel in Discord to send out any messages to them, including their summer tasks. You can also see my previous messages regarding their onboarding tasks, but I'll list them here too:

- [Reading Agile/Scrum basics](#): This needs to be updated to more accurately reflect the development lifecycle at Blueprint and not just agile/scrum.

- [Frontend/Backend roadmaps](#): I can give you access to these if you want to change anything, but they just cover basic articles for things that they might use. We treat all developers as full stack developers now, so reading both instead of just one would be a good idea.

- [Git/GitHub challenge](#): A small exercise to confirm that developers know Git and GitHub, and won't cause major headaches down the line (you're free to test this out if you'd like).

- We used to have a developer challenge (similar to the application challenge) but I don't think this is necessary anymore. Instead, it would be nice to do some workshops or challenges with AWS (basic IAM permissions, deploying Lambdas, etc), system design, and AI tooling (CLI agents, our internal AI tooling, etc). This could possibly be done during the in-person onboarding session, or as a summer assignment via GitHub Classroom or any other tool.

- Project lead onboarding

- As we no longer have product managers, and instead have project leads, they will require a new type of onboarding. Unfortunately, I don't have anything prepared for you :) But I'm confident that both you and Alexis have the knowledge to help them onboard for the upcoming year. I would recommend doing minimal onboarding (familiarity with AWS/deployment pipelines, reading over previous architecture docs, planning sprints, etc) and actively helping them through the first month or so (writing architecture docs, semester timeline planning, writing initial tickets, coordinating with

design team, etc).

May

- Project handoff

- Discuss with VP Engineering/President the optimal way to go about handing off the project to the NPO. This can include things like deployment options for the nonprofit and ensuring proper documentation/repository handoff.

- Review NPO applications

- This will happen in Late May, so make sure to follow up with any potential nonprofits if they have not submitted their project application yet. Currently, Greg has said that they will have another, more complex project that he will send a proposal for, but has also invited us to make a post on the OR forum to seek new projects. I think this will be extremely important! Hopefully there will be enough interest in the community for 4 projects. If, for some reason, there aren't enough projects, then you'll have to work with Pranav (new VP Engineering) to see what kind of internal tooling project(s) you guys can come up with. However, I think there is a lot of need for our services, it just comes down to what kinds of projects suit our organization best. Speaking of which...

- I sent out a [returning developer form](#) a while ago to gauge what developers might want to do for next year. A lot of people said agentic AI, so I would look for projects that might include that. In general, projects that go beyond simple web apps are going to be the most engaging. Good examples of this include Open Referral, Sarapis, and NYCMesh, all of which you can find documentation for in the Projects folder in Google Drive. Absolutely NO CMS work! We have done a lot of CMS projects in the past and they are extremely boring and mundane.

June

- Finalize list of NPOs

- Not much needs to be said. Narrow down the list of NPOs and send out emails to them with an acceptance/rejection.

- Send out project team preference form

- Once the list of nonprofits is finalized, send out a project team preference form to the project leads and developers. This is just a first come, first serve basis. Then, team members will be assigned roles in Discord for their respective teams, and project leads will choose a senior developer.

July

- Project lead/senior developer plan out architecture

- As mentioned before, I would recommend you and Alexis to help them out with this. Basically, the goal is to have an architecture draft that can be shown to the PoC to get feedback on during the initial meeting.

- Initial meetings between project lead/senior developer & NPO

- The PL/SD should meet with the nonprofit at least once in July to meet the points of contact, and get an idea of exactly what they're looking for.

August

- Finalize architecture, set up environments, write initial tasks

- While staying in regular contact with the nonprofit, the PL/SD should finalize the architecture,

set up all production and test environments on AWS and GitHub, and write initial tasks for developers. I would recommend going to the first meeting of each team in September to introduce yourself and make sure that everything is on track.

October

- One-on-one project lead meetings
 - Whether or not you want to run these is up to you. I liked to hold them monthly, but there might be a better format for this.

November/December

- Dev onboarding (fall application cycle)
 - Same as spring onboarding, except no new project leads.
- 1 on 1 dev meetings
 - Again, up to you to decide if you want to hold these. I liked to do these once a semester, and got some pretty valuable feedback each time.

January

- Project lead/senior developer plan out spring semester
 - Same as July/August planning. Should be done during winter break (start a few days after the new year).

March

- Returning developers & project leads
 - Send out forms for returning developers and project leads. We got lucky this year, and we have exactly 4 project leads, but if there is more/less interest in the role, then you may have to do some interviews or convince people to apply.

Reflections

Here are some of my reflections from the previous year:

- We have historically had an extremely low percentage of our developers be female. Now that we have a lot more women on the executive board, I think this will be less of a problem, but I know that there are talented women in CS out there :) Perhaps partnering with SWiCS or reaching out specifically to women in any eboard member's CS classes might be a good way to solve this.
- Not many people know how to run/structure meetings, so giving them some guidance is great. I like to start with a brief status roundtable, then go into deep dive topics (which I have prepared before the meeting) and discuss any blockers that might come up, all while taking meeting notes/minutes. There should be templates for this (and everything mentioned in this doc) in the Google Drive. Personally, I hate long meetings, so I try to only discuss or do things that are important to do with everyone there in-person.
- Give project leads some flexibility on how they want to run their teams. This past year was the first time we had any kind of actual structure, so I would often tell the project leads how to run their teams. However, as the structure has come together over time, I realized that people had different methods of leadership that came to the same result. If I had to do this again, I would give them the tips and tools but not force them to use it.
- However, I really do think that sprint planning is the way to go for project management. I think this would be helpful from a developer perspective as well, as it gives them more insight into the

planning (at the beginning of the spring) versus having a constant stream of tasks that are created by the project lead. Developers have overwhelmingly said that they want to have more product ownership, and would like more DevOps and system design skills.

- The ideal workload is around 3-5 hours per week. A good way to gauge this is simply by asking developers, and then following up with the project lead to discuss potential changes if it is too much/too little work.

As a final note, VP Projects is there to identify and make large structural changes to the way project teams are run, not to actively contribute to or make decisions for the teams. I am sure you understand this already though, and I have no doubt that this upcoming year will be the best experience for project team members in Blueprint history.

Brandon

Revision #2

Created 2 April 2026 14:35:20 by Blueprint Admin

Updated 9 April 2026 21:39:54 by Brandon Yen