opensource.com

A 10-step guide for a successful hackathon

by Tiffany Long

A 10-step guide for a successful hackathon

Hackathons are easy. How much thought do you need to put into them anyway? Just set a date, and people will show up. Well, that is not quite true!

While you may get lucky with that approach, the reality is that hackathons are a keystone experience in the tech industry, and attendees have specific expectations. Not only that, but your organization also has certain needs and should set goals for a hackathon. So, how do you ensure that a hackathon works for your organization and attendees?

A successful hackathon depends on several decisions that tend to be recursive. Decisions about what you want to achieve will impact what resources you allot and how you want to communicate. Those decisions affect whether you go virtual or in person, and that decision will once again impact the resources you need and how you communicate. Alignment when planning hackathons is not just about getting people to agree. You will have a whole suite of decisions that must internally align. For example, a technically difficult hackathon might not be able to attract a large audience (ask me how I know!) and will require a specialized recruitment strategy that requires different resources.

I've done many hackathons over the years, including just a few months back, when my organization hosted a hackathon that led to new features that we will incorporate into the next version of our open source product, Traefik Proxy 3.0. So, trust me when I say planning a hackathon that will enrich attendees and create valuable outcomes for your project is about more than hope, pizza, and chaos.

This article uses the most recent Traefik Labs Hackathon as a blueprint. I share a checklist, tips, and tricks to help you identify your objectives, plan, manage the contest and compensation, share your results, and manage the long tail of the hackathon (the work isn't over when the hackathon ends!)

This guide serves as a model for you to outline best practices so that you, too, can hold a successful hackathon with a sizable target audience that delivers results!

1. Three questions to determine your goals

The first and most crucial step is to set your goals. But this is no simple affair. Before you set goals, you need to coordinate internally on multiple fronts and ask questions such as:

- Why do you want to do a hackathon?
- Who do you want to attend?
- How are you going to measure your success?

Identify your internal stakeholders and set expectations

Hackathons are cross-functional. No hackathon is run by a community person alone. It is important to ensure everyone is aligned on the goals, what is required to achieve them, and that the necessary resources are committed. This probably sounds super corporate, but these functions exist even within the smallest projects. A project needs adoption and code. It also needs value decisions based on who will be using it. And, of course, projects need passionate contributors.

Hackathons require cross-functional resources. One team with a single set of resources cannot successfully run a hackathon. The organization must make various resources available, including:

- Marketing for planning and outreach.
- Product Management for product and industry-specific insight.
- Engineering for deep technical knowledge and community engagement.

For these reasons, hackathons usually support cross-functional goals. Your Community Team, for example, might want to build ownership and convert users to active community members. The Marketing Team might want to enhance awareness and court new users. The Engineering Team might need new perspectives on specific needs or challenges. The Product Team might have goals or no-go areas the community should be aware of.

And last but not least, the hackathon budget is cross-functional. I am sorry to inform you, but hackathons ain't free! Your largest expense is always the dedicated time of your team.

2. Why are you doing this?

Setting your goals is the most important part of a successful hackathon. If you don't know what you want to do or why a hackathon is important, at best, it will have a ton of wasted potential and be a disconnected mess at worst.

Communities feed off of ownership. Decide what you need from your community and what ownership stake you want community members to have. Without a clear understanding of this, your hackathon might not reach its full potential in empowering your community.

Be very careful with your hackathon design and goals. Different types of hackathons appeal to different skill levels. If the code you're looking for is very advanced, take the extra time to court the right audience and accept that it will include less overall attendance. Cast a wide net if the contributions can vary in skill and experience.

Are you hosting a hackathon to get code and build your project?

- Sometimes, projects hit a critical juncture or acquire a lot of excitement around them, and you want to harness the energy to build something together. A hackathon is a great way to achieve this!
- If you have an active community of users, a hackathon can bring everyone together at the same time to harness that excitement to feed the creative energy of your group.

Note: This is more easily achievable with smaller groups who know each other and have a shared experience with the project. You also need to carefully evaluate the skills required to build your project.

Are you hosting a hackathon to build your community or re-engage them?

- Maybe you are just building your community or noticed that your community needs a little juice. Hackathons are exciting, and they can help bring that back.
- Above, I said, "Communities feed off of ownership." If your community members do not feel they have a stake or that their needs and voices matter, they will drift away. This is common when projects grow and become more formalized. As the barrier to entry rises, the ability for community members to feel ownership falls, and the project becomes like a product to the user. One way to enhance community membership is by

creating events that engage users and lower the bar for entry: Bug round-ups, light requests, and longer timelines.

- Perhaps your user community is growing, but the contributor community is becoming more specialized as your tech becomes more complex. In this case, you need to court sophisticated technologists who understand your tech and the use cases. Look for community members who use your tech in their jobs—especially at companies with large or complex deployments. These people are more likely to understand the needs of users and of the tech itself. They will also have suggestions for significant and valuable enhancements.
- You are free to choose goals that build your community and match your team and community members' energy and time. For example, at Traefik Labs, a hackathon aimed at enthusiastic folks with a small time commitment might target our Plugin Catalog. However, when looking for larger contributions or contributions that take significant expertise, we might target advanced technologists–especially those we know.

Are you hosting a hackathon to celebrate something?

- Hackathons are a great way to celebrate a new launch and hype your community. For example, that is exactly why we hosted the Traefik Proxy 3.0 Hackaethon.
- Hackathons are also great for getting the word out about a new product capability. The Traefik Plugin Hackaethon is an excellent example here.
- Maybe you want to organize an event to celebrate your top contributors. Do it with a hackathon! Take a look at this hackathon organized by HackerOne. And if you're thinking, "but this is not about open source software (OSS), how can it be a hackathon?" I've got news for you—hackathons are not just for OSS! Hackathons are for creating with a large community.

Are you hosting a hackathon to build awareness?

Hackathons are a great place to begin if you are just starting and want to build awareness around your product/brand. However, there are a few conditions.

- Laser-focused goals and big contributions are unlikely to happen at this stage. Go for a softer, broader focus, and minimize the work required by attendees.
- Reach out to new community members, less experienced users, and users with less exposure to your specific project.

Are you hosting a hackathon to connect to users?

I can think of no better way to connect new users to your project than a hackathon. Not only will your users become intimately familiar with your project, but hackathons also have a unique way of engendering a sense of ownership, rarely achievable through other types of events.

3. Who is your audience?

Assuming you have pinpointed why you want to host a hackathon and what you want to achieve, it's time to assess the characteristics that a participant needs to be successful. Use your decisions about your goals to identify your audience to ask what type of community member can help you achieve your objectives. Use the list of comparisons below:

- Highly-skilled vs. mixed-skilled vs. low-skilled
- Specialized vs. generalized skill
- Intensive time vs. less intensive time
- Individual contributions vs. group contributions

Your most active community members must look a bit like your target audience.

You might rethink your goals if your target audience doesn't align with at least 80% of the people you know you can attract. Accurately identifying your target audience will go a long way to making your communication strategy around the hackathon and the hackathon itself more successful.

4. How are you measuring goals?

Perfect, now that you answered the first two big questions and have your goals laid down, it's time for the third big question—how will you measure those goals? Inspiring your internal teams and your community to work together in building the future of your project, engendering ownership, and increasing engagement are awesome, but you can't determine success if you can't measure your goals.

What does success look like immediately after the event?

- Clearly, the first sign of success is that your attendees had an overall good experience and are motivated to engage more with your project.
- If you are looking for outreach, set a quantity of participants to shoot for and a number of participants who return to contribute more after the event or in three months.

    - A major sign of success is whether attendees connect and engage with each other, co-educate, and build teams during their hackathon.
    - Were mentorships built? Through partnership, did several newer users grow into skilled mid-level users, or did mid-level users evolve into expert-tier users? This is the gold ring of success indicators.
    - Did your partner organizations (maybe universities) request future hackathons or other events?
- If building awareness, you might also look for successful follow-up chatter. Who wrote blog posts? Were attendees talking about it on social media?
- If you are looking for contributions, did they work for you? Are these the contributions you want? Did they impact how your team thinks about the problems they face? Will you have ongoing collaborations with these contributors?

What will denote success three months after the event?

Defining benchmarks for long-term success is just as important. Here are a few examples of what could indicate long-term success:

- Your hackathon should increase the number of returning contributors to your project. The goal is to get people hooked. If people new to your project came from the hackathon and stayed as users, or if your existing users became more active, you know you won.
- Hackathons are great as self-contained events, but they are supremely valuable as marketing content. They build trust in the community, showing you are responsive and value community input. They are fun loci of activity that let community members bond and look forward to the future, and they are aspirational. People love to see others celebrated and plan to achieve that celebration in the future.
- When you build marketing content around your hackathon (or better yet, others build content around your hackathon), you can expand your reach among second-degree connections.
- Tall poppy syndrome is a shame. Hackathons are a great opportunity to gather those participants who stood out and galvanize them to do other cool things and spread the word about your project.

5. Decide on in-person vs. virtual

I know what you're thinking—is in-person even a consideration? We've all gotten so used to doing everything virtually in the post-covid world. So, are the days of in-person gone? I would argue no, they are not. With care and safety measures in place, in-person events are the heart and soul of hackathons.

- In-person means no distractions, lots of pizza, and energy drink-fueled friendship.
- In-person fuels group participation rather than individual contributor participation.
- In-person works well at scale and in miniature: Organizing in-person hackathons for large groups brings high energy and rewards. But they can get quite costly. If you want to organize a large-scale hackathon, you'll be more successful if you target less experienced developers (students, clubs, new careerists) because these folks have the most time and the most to gain when demonstrating their skill and passion.
- In-person also works well for small groups and is great for intense planning and iteration—long nights with new and old friends, usually over food and beer!

And while many pros come with in-person hackathons, it doesn't mean that the virtual experience only comes with cons. Granted, nothing replaces that feeling of late nights with pizza, off-the-cuff remarks that end up changing your entire project, and a friendly set of eyes over your shoulder as you test or debug. But...

- Virtual means you can get a wider group of participants at a significantly lower cost.
- Virtual respects disability.
- Virtual is geolocation friendly.
- Virtual allows for higher individual contributor participation.
- Virtual offers more flexibility in the style and length of the event – you cannot have a month-long in-person event!

Timelines of virtual hackathons

Did you decide to do a virtual hackathon? Great! Now, you need to determine the type of virtual hackathon you want. Do you envision a prolonged or intensive timeline? Keep in mind that the type of virtual hackathon you choose will determine, to some extent, your target audience and communication strategy.

Extended timeline:

- Allows after-hours tinkering and enables developers to attend without taking time off from work.

- Provides more time to solicit contributions.
- Requires fewer resources for both the organizer and the participants.
- Extended timelines require fewer real-time resources.

Intense timeline:

- Recreates that feeling of intensity usually experienced in in-person hackathons.
- Requires a high amount of resources for a short period of time.
- Requires tight management and a communication platform.
- Requires clear one-on-one communication, but also fosters group-to-group or intra-community communication.

6. Build your communication strategy

Speaking of communication, once you have your goals, you must decide **who** communicates with participants and **how**. It's common to choose between the popular apps of the day. Your choice impacts the event's feel. Different chat applications and collaboration platforms have their own cultures and strengths. The decision you made early on about how to host your hackathon (in-person or virtual, prolonged or intense timeline) is likely to have the most significant impact on your communication strategy.

In-person communication plan

If you are running an in-person hackathon, consider it a genuine event—it feels almost like a conference. In-person hackathons often include the following:

- **Workshops/round tables:** Meant to educate and develop new industry standards/best practices for the concerns of the day. These sessions can function as proctored time-bound conversations amongst 6-10 individuals, where they agree upon findings and take notes that are made public to all participants.
- **Planning sessions:** Often used for projects with non-code outcomes, like developing updated standards.
- **Coding sessions:** Used for code-based projects which require work to maintain and enhance.

Each of the above has different communication needs:

- General communication for running the event (food, cleaning, management of resources).
- Masters of ceremonies to move through the agendas.

- For workshops:
  - People prepared to facilitate, but not lead, conversations in workshops.
  - Note takers and people to make sure that the notes are turned into a publishable product.
  - Project managers to ensure the above tasks are done.

Making this all happen requires the resources and specialized knowledge from your Community, Product Managers, and teach-savvy teams. From past experience, it took a team of community members and staff to manage an event of this scope. To be successful, your team will need specialized people as well.

You also need to decide what types of communication you want to foster and who is responsible for it:

- Multiple teams will need to take shifts to perform general support.
- A DevRel, engineering, or support team will need to manage technical communication between triage and participants.
- Community Teams usually spend extensive time connecting participants to help build strong groups by reinforcing skills or points of view. This is one way to ensure that hackathon magic.
- Community Teams also need to support marketing efforts to engage participants and manage follow-up.

Virtual communication plan

For virtual hackathons, the choice of a communication platform depends heavily on the outcome you want to achieve, the timeline you've chosen for your hackathon (prolonged or intensive), and the type of communication you wish to facilitate (synchronous or asynchronous).

Using Pull Requests and Issues on Git hosts (asynchronous):

- Choosing to communicate through Git pull requests and Issues on your project directly frees up technical staff resources because it keeps the conversations about projects in your current process and allows your team to be responsive rather than instigating communication.
- This approach is great if the team for your hackathon is small or if the expected contributions are relatively small and you do not plan to help participants form teams.

- Using your existing processes is especially great for prolonged hackathons as they do not require additional moderation or require your team to monitor an additional app.
- The downside is that you will only facilitate communication with the individual contributor or group of contributors already working together. It's difficult to connect participants who are working separately. Participants can't find each other as easily on their own, so you lose some of the magic that happens when hackathon participants organically talk to each other in open threads.

Using a chat application (synchronous):

- Choosing dedicated chat servers is required for intense hackathons.
- Chat facilitates the team formation and communication necessary for complex projects with fast timelines and sparks the brainstorming that preludes an awesome contribution.
- Additionally, your goal is to build community. People want to join communities where they have ownership, have friends, and feel comfortable. You need a place for participants to feel connected to each other if you want them to return.
- Chat servers can outlast an event, allowing for continued community engagement.

Regardless of which platform you choose, you need a communication plan that identifies when every person on your team is available. Managing a virtual hackathon can get quite tricky, primarily due to the different timezones—people can participate whenever they want, from wherever they want. You must plan to accommodate participants across all time zones and for every occasion. Draw up a plan with who is responsible (and when) for the following:

- Determining response SLAs.
- Animating your virtual space (a dead space guarantees poor communication).
- Encouraging team building.
- Responding to technical questions.
- Checking in on participants.
- Moderating the space to ensure the safety of your participants.

7. Decide on the prizes

Is your hackathon a contest? Hackathon participants are often content with grand prizes and "swagpaloozas" for top contributions. But before you decide on the fun stuff (the actual awards), you must determine what your contest values.

- What differentiates a good contribution from a great contribution? If your attendees know how you feel about this, they are more likely to hit it out of the park.
- What do you value? This is your chance to tell participants what you want to see submitted by attaching a prize to it. For example, during the last Traefik Hackaethon, we offered bounties for the most-wanted features. These were, indeed, the ones most people worked on.
- Are there categories of contributions? You need to decide on prizes for each category.
- Create a rubric (a chart or grid defining and ranking achievements, like this example). This way, participants know what you value and how they are judged. This was one way we improved submissions at HackerOne.

On the other hand, some may argue that competition is overrated. If your goal is participation, feel free to reward every single one of your participants for simply giving back! Hacktoberfest is a great example of this approach.

8. Swag it up

Everyone loves swag! And your participants would certainly appreciate a token to remember this event, whether virtual or in person. Swag has two purposes:

- Swag shows your appreciation: The contributors took their time to engage with you in an intense way; thank them with a gift that shows you value their contributions.
- Swag builds awareness: Gifting swag to your participants helps them spread the love and build awareness of your community by sharing their loot and experience.

The community loves swag, but they don't love boring swag! You probably distributed your existing t-shirts and stickers during another event. Make your hackathon memorable and go for new, exciting, and exclusive designs. Shirts are great, and hoodies reign supreme. But think about cool swag participants may not have already. Think of something that could become their new staple item, like backup batteries or hats (both popular at HackerOne). Personally, my own home features some towels and slippers from hackathons!

9. Get the word out

Setting your goals and deciding on amazing grand prizes and swag are all important steps. But how will anyone know your hackathon is happening if you don't get the word out? You need to investigate the available channels carefully, and you need to be bold with your promotion. I'm talking blogs, vlogs, emails, social media—anything you can get your hands on.

However, depending on your defined goals, you need to invest in the appropriate channel. Where you advertise depends on who you want to invite to your hackathon.

- IIf you want to attract more experienced users, target big organizations where your project is used. LinkedIn and email promotion would be most effective here.
- If you want to bring in new and less experienced users, you're better off targeting universities and boot camps. Promoting the event on community-based media, like Mastodon, Matrix, Mattermost, Reddit, Discourse, Discord, and any place your target audience hangs out is a better choice.

10. Managing the long tail

Yay, the hackathon is over! Now all hackathon-related activities can stop, and we no longer need to pull resources, right? Wrong! Think of hackathons as only one step of the road in a series of events in your software development and community building. To deem your hackathon a success, you must be prepared to engage in post-event activities.

- Communicating your results: Don't forget to communicate hackathon outcomes internally and externally. Demonstrate the ownership the community members gained during the hackathon to grow trust in your community and project.
- Building community: Lean on your hackathon participants for future community activity.
- Putting together the retrospective: What went well, what went terrible, what was meh, what surprised you? This analysis is how you grow, change, and iterate. Don't forget to do a blameless retro as soon as possible so it is all fresh in your mind.

Wrap up

If you started reading this article thinking that hackathons aren't that hard to pull off, I'm sorry to have burst your bubble! And although I sincerely believe hackathons are a great way to engage and communicate with your community on so many levels, having just the intention does not guarantee the results.

For a hackathon to be successful, you need to be meticulous and prepared to invest significant resources and effort to plan and execute it properly.

Thank you for reading, and I hope this checklist helps you successfully organize your next hackathon!