Using Ongoing Hackathons to Accelerate Digital Transformation: A Practical Handbook

Application leaders increasingly use hackathons as a key element of ongoing business delivery, innovation and digital transformation. While many hackathons underachieve, the most successful follow a predictable formula, focusing on learning, culture and results before, during, and after the event.

Key Challenges

- A lot of hackathons are not designed to achieve specific target outcomes. In fact, they are not designed at all; they are just run to see what happens next. It's not surprising to see that they fail to be of much use.

- A number of high-potential hackathons never happen because the business sponsors are reluctant or unconvinced to follow through winning proposals, or legal hurdles stand in the way.

- Hackathons are often treated as stand-alone events, which makes it difficult to prove their value within long-term digital transformation efforts.

- Running a hackathon to appear innovative will have only marginal impact nowadays because so many companies are doing it already.
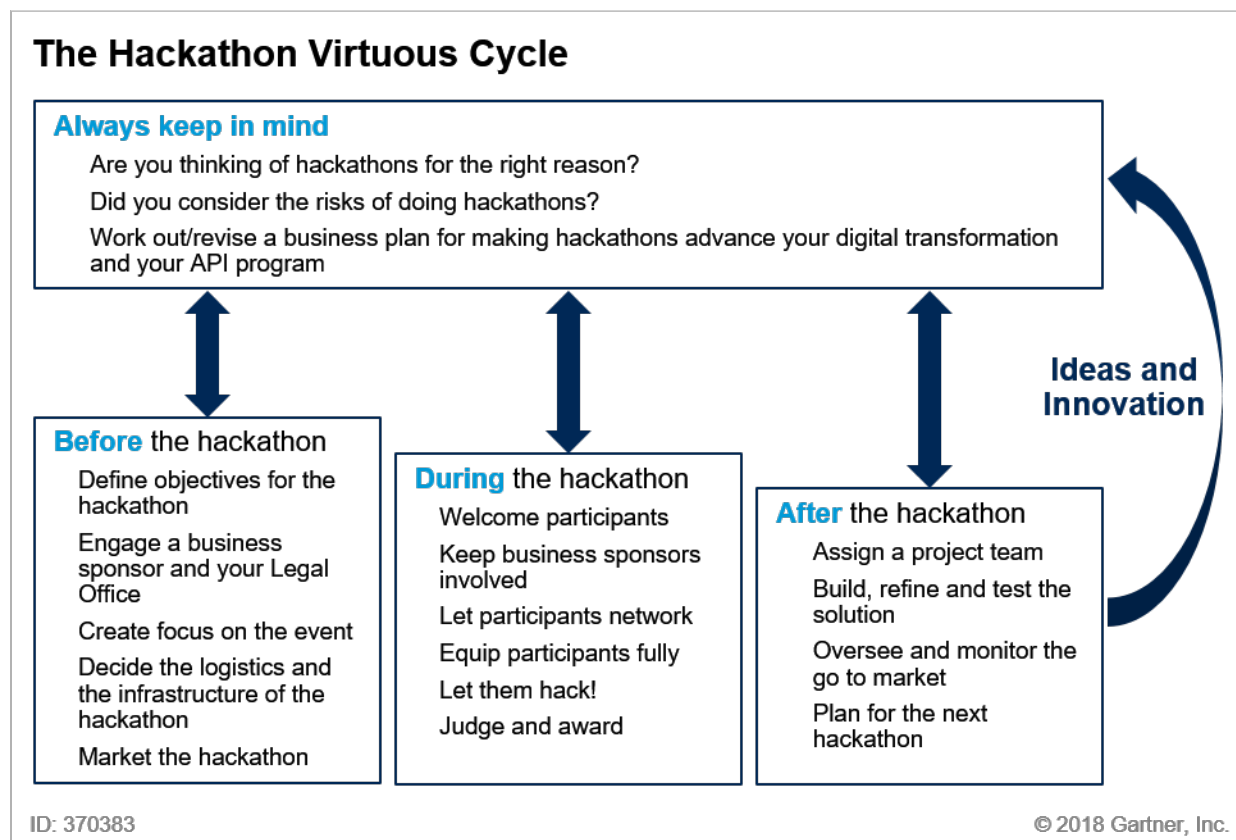
Recommendations

Application leaders responsible for application development strategies for digital business should:

- Make hackathons an element of a digital transformation and business delivery that you *always keep in mind* rather than one-off events, by defining strategic objectives and creating specific goals for each hackathon.

- Allow sufficient time to properly prepare *before you run a hackathon,* 3-6 months on average. (The legal arrangements around intellectual property, the focus of the event on a specific business objective, and the marketing of the hackathon are the most time-intensive tasks.)

- Organize the way you *run a hackathon* according to local culture, the expectations of the business sponsors, the profile of the participants, and the rewards the participants expect.

- *After the hackathon,* assign a dedicated agile team to bring the winning ideas and experiments to fruition and ensure they are ethical, reliable, secure and compliant. Include at least the winning team, a product manager and a business domain expert (who might become the product owner).

Figure 1. The Hackathon Virtuous Circle

The Hackathon Virtuous Cycle

Always keep in mind
- Are you thinking of hackathons for the right reason?
- Did you consider the risks of doing hackathons?
- Work out/revise a business plan for making hackathons advance your digital transformation and your API program

Ideas and Innovation

Before the hackathon
- Define objectives for the hackathon
- Engage a business sponsor and your Legal Office
- Create focus on the event
- Decide the logistics and the infrastructure of the hackathon
- Market the hackathon

During the hackathon
- Welcome participants
- Keep business sponsors involved
- Let participants network
- Equip participants fully
- Let them hack!
- Judge and award

After the hackathon
- Assign a project team
- Build, refine and test the solution
- Oversee and monitor the go to market
- Plan for the next hackathon

ID: 370383

© 2018 Gartner, Inc.

Source: Gartner (December 2018)

Introduction

A hackathon is an event at which developers, designers and subject matter experts collaborate intensively, developing ideas beyond their standard work, on software (or hardware) projects. The goal of a hackathon is to create at least parts of usable solutions and develop ideas. Organizations often host hackathons to offer a less-structured opportunity to discover creative and innovative uses of their technologies. Hackathons tend to have a specific focus, which can include the programming language used, an application, an API or an industry category, and generally end with prizes for winning ideas. Ideally,

> The end of a hackathon is the beginning of a new business idea that is worth testing the market for, and presents opportunities for new hackathons.

There are many types of hackathons. They might be internal or external; virtual or physical; distributed, colocated or in person. They might be short (a weekend) or long (several weeks). There are ways to run hackathons to suit the culture of your company, or your local culture, or the working style of the developers, designers and subject matter experts the hackathon targets.

A quick look to hackathon.com, AngelHack or hackalist.org makes you realize that hackathons are run every day, worldwide, by companies in all verticals and government institutions. This is a phenomenon that no company undergoing digital transformation or implementing an API program can afford to ignore.

Many hackathons are happening around the globe, but a lot of them don't deliver tangible results, because they have been poorly chartered and executed. A lot of good ideas are left dangling, and opportunities are lost. So how can hackathons meaningfully contribute to business objectives and advance API programs?

This research contains a lot of detail about spinning within the hackathons' virtuous circle (see Figure 1). It will show application leaders what to do, how to create innovative value, and how to make hackathons work for a business purpose. Like any change initiative, hackathons require thoughtful planning and execution.

The upper block in the figure above lists three things that you should consider before even thinking of doing hackathons, and those things that you should *always keep in mind.* You should also take the time to revisit them once you are finished with the preparations for a hackathon, and once the hackathon has just ended (after the "Before" and "During" blocks).

What happens after the hackathon should be considered as input for the hackathons to come, because it might change their focus, potentially in a very significant way. That is the nature of the beast, and you should welcome that.

> Hackathons are a great way to test ideas and create opportunities for innovation because they are disruptive in nature.

Making internal and external hackathons an ongoing part of a digital transformation effort is one way for application leaders to deliver changes to their application portfolios faster, cheaper and better. Facebook is an example of an enterprise that uses ongoing hackathons. Its first official internal hackathon was in 2007,[1] and the company continues to run them approximately every six

weeks, or whenever anyone decides to have one. Video, the Like button, Chat, and the ubiquitous Timeline were all created through internal Facebook hackathons.

Application leaders should strive to use hackathons to discover and deliver real-world business value today and beyond. For this reason, Gartner has developed a comprehensive set of best practices for running a truly great hackathon. The focus of this research is best practices that will enable application leaders to use ongoing hackathons to accelerate digital transformation.

Analysis

Things to Always Keep in Mind

As previously noted, this section contains considerations that you should keep in mind before even thinking of doing hackathons, before the "Things to Do Before You Run a Hackathon" (yes … before the before).

The section below outlines basic rules, and you should surely take the time to revisit them, and possibly adjust them, at specific times:

- Once you are finished with the preparations for a hackathon (after the "before").

- At the end of the hackathon (after the "during").

- When the results of a previous hackathon start hitting the market, and you are ready to start planning the next hackathon (after the "after").

Possible Net Profits Coming From Hackathons

- **Reducing time and cost to market:** Hackathons can be a highly cost-effective means of prototyping new channels and mobile apps. For example, a Facebook intern built the "tagging in comments" functionality at a hackathon. The company brought this to market for 100% of users within two weeks. Gartner has seen some application leaders reduce the time to market for new capabilities by as much as 90% using this technique.

- **Identifying new revenue or service opportunities:** Application leaders that broaden their constituency and scope can also use hackathons to create proposals or ideas for new types of products or services, or even new business models. These ideas might just be testing new business opportunities, such as new products or offerings. In this case, they provide a much quicker and clearer answer than months of internal discussions, where different internal constituencies might have totally different views on the potential return of a new idea.

- **Engaging new partners and jump-starting new partner ecosystems:** Hackathons are actually a community-building exercise as well, and should not be viewed as just an event. Outsiders (like students from a university, for example) could provide valuable external points of view. Many participants look at hackathons as a way to hang around with like-minded peers, or learn how to build a platform themselves, and would come back hackathon after hackathon.

User Experience Benefits

- **Improving employee experience:** Organizations that are implementing digital workplace initiatives are exploring ways to encourage greater participation from all employees. Hackathons that bring business and IT employees together can foster unique avenues for learning, while all staff take advantage of the opportunity to either improve or demonstrate their digital dexterity. Additionally, internal hackathons give employees a sense of ownership over their work practices and allow them to network with people in the organization they might not otherwise have the opportunity to collaborate with. Recognition of people's achievements and innovation within these experiences can change an organization's culture, and help new employees to become better-known. It can also invigorate senior staff who might feel they have "plateaued." In this sense, internal hackathons become a cultural barometer with which business and HR leaders can gauge employee engagement and organizational identity (brand). Indeed, HR leaders may want to position internal hackathons as a means of attracting new internal hires for sought-after positions.

- **Improving the customer experience:** Hackathons that accomplish this are often focused on developers creating new types of customer interactions, through the usage of new devices. Lately, this has been achieved by applying experimental AI technology, like natural language processing, image recognition, or chatbots. After all, digital transformation are about identifying new business moments in which your services could be useful, but are not there yet. Frequently, external developers are also customers of your company, and can often have great ideas about how to improve user experience because of that. They have no reason to hide their dissatisfaction with existing customer experience, and they can do something to improve it.

Internal Operations Gains

- **Changing culture:** Change is particularly important as it encourages much-needed innovation, and can help to shift the focus on what is really important for a company. This is because hackathons generate new ideas and new ways of thinking. Not every idea is a good one. In fact, most of them are not, but the process of coming up with ideas can stimulate new thinking and new ways of doing things — which ultimately can help change culture. Hackathons can help catalyze and reward a new "do it yourself" or "maker" work ethic as part of cultural change.

- **Funding:** Hackathons can help application leaders get the funding needed to bring new solutions to market, if they engage business stakeholders prior to the hackathon. For example, the application leader and the business unit can identify urgent business needs. The hackathon could be structured around these needs, with the business committing to fund further development of the winning proposals. A joint business-IT project team could be formed to bring the proposals to market jointly with the winning team. Mentoring can also be provided, as another way to fuse innovation into the organization.

- **Reducing app design and maintenance costs:** Hackathons can be a highly cost-effective means of developing new channels and mobile app. We have seen some banking application leaders reduce the time to market for new mobile apps by as much as 90% using this technique, and halving their maintenance costs.

- **Attracting and retaining talent:** Hackathons represent an effective way of identifying and attracting new talent, as they are becoming more and more common on college campuses. However, you should take into account that external developers just want to invent the latest killer app, and then move on the next one. They are not necessarily looking for a steady job in an established corporation. It's a culture they would struggle to fit in. Hackathons can also help enterprises retain key talent by providing a stimulating environment where participants have constant opportunities to submit new ideas and solutions to business problems. They can help employees develop new skills and get familiar with new technologies in a safe learning environment.

Unfortunately, we have not seen many application leaders design hackathons with one or more of those goals in mind as yet. Hackathons are becoming mainstream, as the position of the Hackathons innovation profile in a number of Gartner Hype Cycles shows, so companies frequently run hackathons just because a lot of their competitors do. When hackathons deliver good results, like an idea that has concrete potential, they are not prepared (or able) to take it forward. In other cases, a number of application leaders have shared that they are primarily running a hackathon because it is a great way to "appear" innovative. However, running a hackathon to appear innovative in 2018 and beyond is no longer viable, because so many companies worldwide are doing it already.

Risks of Running Hackathons

The primary risks of hackathons are:

- **Failure to deliver business value:** The stated objective of many hackathons is to stimulate new ideas. But this focus is too broad to allow application leaders to drive any real business impact (such as increasing revenue, decreasing costs, improving customer experience, or making a difference to a digital transformation). Driving business value comes from bringing the best ideas to production after the hackathon, frequently within a fixed timeline, and set by competitive pressure. Business value objectives for internal hackathons should not be as rigid. Using hackathons to enable digital dexterity, talent development or employee satisfaction will ultimately create business value for the enterprise.

- **Reputation risk:** Poor hackathon execution can lead to reputation damage. Internal hackathons can be (and very often should be) less-structured and less-formal in order to maximize the creative potential of employees, or to prepare them for an external hackathon (by testing an API layer, for example). But external hackathons need to be run like a world-class event. Poor execution (technical difficulties, for example) can negatively impact brand and reputation. Hackathon participants are in general not shy about letting you know what works, and especially what doesn't. If you prove yourself to be naïve in fixing the issues, the word will spread. This will affect your hackathon plan negatively.

- **Quality variability:** Not every hackathon will result in a game-changing prototype or idea. Risk-taking and failure need to be acceptable — and even encouraged — as part of a hackathon. This is a major issue in many corporate cultures. Innovation comes with risks, and proper attention should focus on the positive lessons learnt when a hackathon does not deliver a killer

idea. Failing fast and learning fast have a lot of value. Networking, skills development and learning are all positive effects, regardless (or even sometime because) of things going wrong.

- **Intellectual property/ownership disputes:** It should be absolutely clear who owns any intellectual property that results from the hackathon, particularly for external events. Legal and risk staff should be highly involved in planning for the hackathon, and create terms and conditions that empower participants as much as possible, yet protect the enterprise. Don't even start a hackathon if this is not in place.

- **Employment risks:** Internal hackathons have the potential to increase employment law risks.[2] For example, questions could arise as to whether or not employee time spent on out-of-hours hackathons is compensable by law. To address this risk, internal hackathons could be limited to employees exempt from overtime, employees could be compensated, or regular work time could be used. This risk goes beyond working hours. There are likely to be issues about rewards (taxation), job descriptions, the definitions of voluntary and involuntary participation in activities beyond a role description, and more, particularly in strongly governed environments.

Develop or Adjust a Business Plan That Makes Hackathons a Key Element of Digital Transformation

Application leaders should create and communicate a business plan for ongoing hackathons that includes the business situation and the proposed solution, as well as the benefits, risks, costs and alternatives. They should also be prepared to adapt the plan. When one hackathon is finished, or at any point in time when you realize that things are not going according to plan (which might not necessarily be bad), look at the plan, and be prepared to adjust it. Maybe the group that is winning the hackathon is not 100% OK with your terms and conditions, or your plans to take the idea forward, for example. Experiment with using different types of hackathon to drive different ends, and sharpen your focus to maximize the value of the next hackathon.

Application leaders who believe that hackathons can enable their organization to accelerate digital transformation should make a business plan that includes the following elements:

- Business situation — for example, application leaders need to deliver faster, cheaper and better but lack adequate talent, skills, funding, culture and business-IT alignment to do so as quickly as CIOs expect.

- Proposed solution/objectives — for example, make hackathons part of ongoing delivery to accelerate digital transformation.

- Costs — for example, costs to run the hackathon and bring winning proposals to market.

- Benefits — for example, reducing time and cost to market, improving employee/customer experience, identifying new revenue opportunities or changing culture.

- Risks — for example, failure to deliver business value, reputation risk, quality variability, intellectual property or employment risks.

- Alternatives — for example, startup incubators, techquisitions,[3] acqui-hiring,[4] self-directed teams or scrums.

You also need to consider what would happen if you don't do hackathons at all — for example, would the customers' perception of your company deteriorate, or would your direct competitors leapfrog you. The purpose of the business plan is to secure executive and line of business (LOB) sponsors as you will need them to be involved in the hackathon virtuous circle (see Figure 1).

Things to Do Before You Run a Hackathon

The first, obvious step toward making hackathons a key part of ongoing delivery is to start running them. Not so fast. The previous section (about things to always keep in mind) contains work that really needs to be done *before* you start running them. Sadly, most companies rush into hackathons, generally because of competitive pressure, and fail to do some basic preparation. The general rule here is that you get out of hackathons as much (if not more) as you put into their organization*, before the event starts.*

Application leaders can begin by identifying the strategic objectives and key challenges of specific LOB executives, and proposing a hackathon to quickly deliver a solution proposal. Start with an executive who has an urgent need to respond to market pressure more quickly and, ideally, is known to be passionate about creating change and trying new things.

The why, what, who, how and when are clearly spelled out in this section for added clarity.

Define Objectives for the Hackathon (the Why and the What)

The first step toward a great hackathon is to align it to the strategic objectives of senior business executives. In other words, you start with a strong why?

Some hackathons are explicitly purposeful: "We are here to do XXX by YYY." These types of hackathons, while they have a specific purpose and a desired outcome, can be easy to justify to decision makers because sponsors see them as directly supporting the business. However, take into account that, for the employees supporting the hackathon, it will more than likely come across as unpaid overtime. Because the structure and goals are already somewhat established, it is also easy to diminish the innovation, brainstorming and other types of positive ideation experiences that employees might be looking for. This type of hackathon may also reduce the community-building aspects, as well as the overall social experience.

At the other extreme are hackathons that are overly social — more of a "digital Woodstock." The sponsors pitch it in such a casual manner that it is difficult to justify. It might also make it difficult to really do the proper support and event prework on data, tools and other components of the hackathon, because the event has little shape or form. We would recommend that you avoid this type of event.

Somewhere in the middle is a challenge-based hackathon with some degree of purpose, some degree of outcome in terms of a theme, but in a very open-ended way, leaving it up to participants to navigate freely.

Failure needs to be absolutely OK, and even expected.

Experimentation should be encouraged. There is no single right answer to the theme. If you don't allow the possibility of failure, you limit innovation. The hackathon has to be open to a wide range of participants, and everyone should know that their contributions are valued. There has to be a social experience that makes it enjoyable, in line with the theme and lighthearted competition. There is a before and after, and there is a community-building dynamic. This approach makes the hackathon not just an event, but a more intriguing journey, and is attractive to developers.

Hackathons organized around a specific challenge help to avoid low-quality and unfocused ideas. Ensure the challenges are clear and testable (as with the Citizens Bank Challenge, for example),[5] but not overly prescriptive, as discussed previously.

Some common examples of challenge-based hackathon objectives are:

- Internal impact by helping senior management think about digital business, innovation and APIs in new ways (via proposals and prototypes generated at the hackathon).

- Reduce the time to market and cost of new mobile apps.

- Investigate the potential of artificial intelligence and other emerging technologies.

- Engage new partners and ecosystems to create added value for the company and its customers by publishing access to data.

- Gather feedback on the company's existing APIs.

Create Focus for the Hackathon (the What)

A "design thinking" approach to a hackathon focuses on the user journey and the personas involved, allowing pain points to be identified and addressed. It also encourages attendees to understand the needs of end users, and to target what they build rather than being only "blue sky" and speculative.

The strategic objectives should then become even more focused on specific business or technology outcomes. These objectives should be focused and phrased in a way that is appealing to developers, designers and business developers. Obscure challenges that are internal and specific to the company should be avoided.

Just a few case study examples include:

- An order-to-cash process in 2020.

- Banking in the age of blockchain.

- Consumables maintenance and the Internet of Things.

- Using public government data to ease transportation.

- How chatbots can be used for customer care.

Gartner recommends restricting the hackathon to no more than two main categories or objectives, because more will likely dilute value creation.

Engage a Business Sponsor and Your Legal Office (the Who)

Prior to the hackathon, application leaders should gain commitment from LOB executives to bring winning proposals to production. Application leaders often do this after the hackathon, when the focus might have changed slightly, and frequently the LOB has, for some reason, become less interested and declines funding. Getting business sponsors to commit to funding winning proposals prior to the hackathon will gain more business involvement and commitment throughout the process. Legal issues in bringing winning proposals to production are quite common, so application leaders should clear the process with their legal offices. This tends to be the slowest prehackathon activity, so it needs to get started very promptly, once the objectives and the scope of the hackathon are clear.

Sometimes the winning hackathon team does not want to bring a proposal further, but your business sponsors might want to — consider this possibility at this stage, with your legal office.

Internal hackathons should not be as formal as external hackathons. Application managers, business leaders and HR leaders should promote the ability of employees to self-form and initiate their own hackathons (though ensuring that they adhere to corporate conduct and other policies). A business sponsor is not necessarily always needed, nor is a formal approach to bringing proposals to production after the internal hackathon. For example, at Facebook, anyone in the company can decide to run a hackathon.

Decide How to Execute the Hackathon (How, When and Some Who)

Some of the key things to consider to execute a great hackathon include:

- **Partners and sponsors.** Many companies run hackathons with a number of partners. Examples of partners and sponsors can be an organization that helps you execute all elements of the hackathon (like AngelHack), vendors that provide tools (see the Tool Selection section), a business or community partner, a sponsor that provides prizes, or a university. Understand what each partner wants and needs, and ensure the hackathon is planned to deliver against these needs. Make sure there is something for everybody.

- **Location and timing.** Will the hackathon be held in a physical location? In some geographies, it could be run from a Friday night through a Sunday morning; in some others, during the week, with proper time for employees to contribute, out of their "normal" working time. It is customary to ease accommodation and provide social events for face to face hackathons. An attractive location for tech-savvy individuals, such as an established meet-up space or innovation lab, will encourage people to participate. Universities can also be an appealing location and would make the hackathon more open and possibly more diverse in terms of gender, race and technology capabilities. If the hackathon is virtual, it can be run over a more extended period

(even several weeks) on cloud infrastructure. A hybrid approach can also work well, with teams beginning their work virtually, but then coming together in a physical location to present the best proposals or prototypes. Plan on locking in the location three to six months before the hackathon.

- **Evaluation criteria.** The criteria for judging the hackathon's outcomes will depend on the strategic objectives and tactical goals that have been set. Criteria should be published in advance of the hackathon, easy for the participants to understand and should speak to them. A few case study examples include:

  - Business potential

  - User experience

  - Creativity

  - Usefulness

  - Functionality

  - Completeness of solution

  - Number of services/APIs mashed together

  - Preference for multivertical user journeys

  - Team that makes most progress during the hackathon

  - Compelling presentation and demo

- **Participants.** The participants in a successful hackathon, and the number of teams they are organized in, may vary widely, depending on the What. For public hackathons, attendees may be drawn from universities and colleges (and from internal staff, customers, partners, technical SIGs/User Groups//MeetUps, "walkin's" driven by promotion/marketing). Complementary services, such as SMS/notifications (for example, Twilio), property valuation services (for example, Zoopla) and social media (for example, Twitter) can also stimulate ideas for composite apps. Cloud providers such as Amazon and Google can offer valuable input and may also provide credit for attendees. Business partners that have been asking for new applications or interactions are an obvious choice. And third-party hackathon organizers can provide participants and sample APIs. For employees that are supporting the event, consider giving them a few days off, given the demanding hours of a hackathon.

- **Tool selection.** Select the tools your company needs to run the hackathon. This, too, will depend on the hackathon's objectives. If the hackathon is focused on coming up with new ideas, no specific technical infrastructure may be required, although a cloud-based prototyping "sandbox" (provided by a sponsoring cloud provider, for example) can be beneficial. If the purpose of the hackathon is to develop a prototype, low code development tools like hpaPaaS could be used, or prototyping tools from Adobe, InVision and others; APIs are almost invariably needed. Many APIs should be stubbed out or "mocked" and not have access to real corporate data. The APIs should be presented in a self-service API developers' portal, normally found in

API management offerings, using a common and easily consumable format. Partner APIs (for example, SMS APIs) should ideally be provided side by side with your company's own APIs. You might also consider including IoT hardware to stimulate hackathon ideas. It is important to note that this developer's experience will be your company's digital face to the public — this has deep implications in industries where the company's image and reputation is of paramount importance (like banking).

- **Technology infrastructure.** Ensure adequate Wi-Fi and storage are available for participants (for example, through a partner like Amazon). Ensure cloud platform capacity and vouchers for user accounts. Provide data that is cleaned up to use, even if it is bogus data. Ensure adequate Wi-Fi, internet bandwidth, power and power strips are available. Provide adequate audio-video equipment, whiteboards, tables and chairs, and other equipment. Ensure the area and infrastructure are accessible to anyone with physical or other limitations. Consider doing a dry run before the event to ensure all processes, tools and infrastructure are working properly.

- **Facilitators, mentors and volunteers.** Any IT support issues will drive pain and frustration among people sometimes giving up their spare time. It is important to have enough mentors available throughout the hackathon to answer participant questions and resolve any issues encountered. You want them to hack, you don't want to put hurdles in their way. A mentor that has subject matter expertise in each challenge area (like "An order to cash process in 2020") should be available to answer questions: partners (see above) could also help facilitate. An experienced facilitator can make the difference between a good hackathon and a great one. Volunteers can help with registration, basic information and questions participants might have. Many clients effectively shut down chunks of the company for the event (except for a few operations and customer-facing people).

- **Demos.** Provide time and equipment for hackathon participants to try dry runs of presenting their demos and ideas. If the hackathon is happening across different sites, provide suitable connection between them. Breakout areas are also very useful. This is one of the most fun and exciting parts of the hackathon for both participants and sponsors.

- **Judging and prizes.** Form a panel of judges who are qualified to evaluate the concepts, ideas or technologies that are developed in the hackathon. Typically, this panel will include internal executives and personnel who have a business interest in the hackathon's focus. But hackathon panels sometimes also include outside industry experts, as well as academic and media figures. In some cases, you might want the participants to have at least a share of the vote. Have multiple categories that recognize and reward creativity, even for demos that aren't specific to your purpose but are great ideas. This is about learning and innovation. Judging should embrace failure — it's okay to fail as long as the participant accomplished something intriguing. Prizes are an important motivation. Cash is always welcome, and so are vouchers for local e-tailers, but developers also tend to be attracted to gadgets, such as drones. Another possibility is offering developer access to mentoring, a working space, take advantage of an incubator program or even travel to a prestigious accelerator. In some cases, the company may want to consider offering the winning solution a share of revenue, or the winning team a job (if a job is what they are looking for).

- **Terms/conditions and code of conduct.** The application manager should work with risk management and legal departments to clearly define and communicate the terms and

conditions under which the hackathon will be held, as well as an appropriate code of conduct. This should be very public, include an overview of the event, stating its purpose and objectives, and define who is eligible to participate and other conditions of participation. *It should be absolutely clear who owns any intellectual property that results from the hackathon,* as well as what constitutes acceptable conduct during the event. Also think about what prework teams are allowed to bring. Should the team that wins any prizes or kudos be the one that makes the most progress, or the one that has the most fully formed solution because it put in some work before the event? Who owns any prework?

- **Emergency plan.** This is too frequently overlooked. Create an emergency plan if an unfortunate event occurs during the hackathon. Make contingencies for factors such as weather, transportation strikes, illness and cancellation of venue.

Market the Hackathon (Who and When)

Three to six months before the event, create a simple, but dedicated website that will market the hackathon to the participants you are targeting, along with a full marketing plan that promotes the event.

The application manager should engage corporate communications and marketing teams' skills early in the process, choosing a social media hashtag and offering a prize to the most active user on Twitter, Facebook, Instagram, Snapchat, and/or WhatsApp. Prepare T-shirts and marketing collaterals to be distributed on the hackathon's first day.

Consider marketing to specific unrepresented groups that are often overlooked. Local technology user groups, meet-ups and developer collaboration platforms, such as GitHub and Stack Overflow, can also be leveraged to get the word out. There are plenty of websites that list upcoming hackathons, like hackathon.com, or hackalist.org.

Leverage your own IT staff and their networks too. Linking the hackathon to an event/conference or using such an event as a promotional channel could also be valuable (for example, present a tech case study and pitch your hackathon).

Things to Do While You Run a Hackathon

Hackathons are a bit like weddings: you spend months preparing them, and on the day you just sit back, let all the things you and your team have meticulously organized unfold, and watch everything fall into place. In most cases, especially if you follow the guidance of the previous section, they do.

Welcome

Kick off the event with a welcome address from the business sponsor, or the CIO and other participating executives. Give a vision of what your digital transformation entails. A third-party speaker can also add context, based on experiences at other hackathons. Senior executives can explain why they wanted to hold a hackathon and how it will help the company, its customers and its community.

The participants should be encouraged to use the hackathon hashtag on social media, and marketing collateral (like T-shirts or any other material) should be distributed.

Provide a clear explanation of the hackathon process. This includes explaining what its objectives are, how it will work and how you'll pick a winner. But don't overdo the speaking. Participants are not there to be lectured. And don't be so prescriptive in the welcome that you take away all the innovation, creativity and sense of ownership you are trying to empower the participants with, and eventually harness from them.

Keep Business Sponsors Involved

The application manager's role during a face-to-face hackathon is mainly to be an executive presence and, potentially, to judge and award. The LOB executives who have agreed to fund further work on a winning proposal should also have an executive presence and vote on their favorite ideas or prototypes. Try to cut through good presentation skills, and look at the substance.

Part of the LOB judging process (whether the hackathon was internal or external and virtual or face to face) will be to ensure that winning proposals are valuable enough. In most cases, a number of solutions will have great potential and value, but business sponsors will decide which ones are best to take further.

The business sponsor should extend offers to the winning proposal teams during the awards ceremony, if possible. Frequently, the offer will be for the winning teams to bring the proposals to market jointly with an enterprise project team. For internal hackathons, the opportunity is for employees to develop their own ideas, which in turn can bring career advancements. For external hackathons, not all winners may be interested in bringing their prototype or proposal to production due to the pressures of their full-time jobs or last-minute disagreement on the conditions required to take it forward. Don't be surprised if this happens, and consider this possibility before running the hackathon. You should also decide what negotiation tactics to apply in this case, and any alternative rewards you can reasonably offer to winning participants."

This is one of many reasons it is so important to provide clarity during the welcome address, and offer a world-class experience during the hackathon itself. Participants who do not have an enjoyable experience are much less likely to want to engage with the enterprise after the hackathon.

Let the Participants Network

For many participants, the opportunity to network with their peers is one of the most-attractive aspects of a hackathon. This can help to attract new talent as well. Social events before the hackathon are encouraged. Executives should be present, in a relaxed and informal way, during the hackathon to show that the company takes innovation seriously.

Provide State-of-the-Art Equipment

The participants will need to be equipped with and skilled to use the various tools that will be available to them during the hackathon. These could include things like the API developers' portal that lists the APIs available for the hackathon (which may include partner APIs and/or the

company's own APIs). Technology partners should be given the opportunity to promote their own APIs, or bring their own IoT devices, as this might spark ideas among the participants.

Let Them Hack

During the hackathon, provide constant support and facilitation, healthy food, fun food (the mythical pizza) and a variety of drinks (not just Red Bull). For hackathons that run during the weekend, let participants bring sleeping bags if they want to, if the venue facilities and security arrangements allow. Provide breaks and checkpoints. If the hackathon is virtual, provide mechanisms for support and questions. Remember that the night before the final day of the hackathon is typically when participants put their work together, and it's not unusual that things don't work — be prepared to assist some of those crises.

Judge and Award

The panel of judges should vote on their favorite ideas or prototypes, in a fair and transparent manner, based on the defined evaluation criteria.

The hackathon should conclude with an award ceremony for the winners. Consider having business partners sponsor specific awards (for example, "Most Innovative App" or "Most Promising Development Approach") and provide prizes associated with them. Technology vendors are often happy to sponsor prizes. For example, an IoT platform provider might sponsor "Best IoT Solution" or a cloud provider might sponsor "Best Cloud Solution."

Things to Do After You Run a Hackathon

Very simply said, capture the wins. All hackathons work, even if no interesting idea came up, the win is that you either invited the wrong people as participants, or the APIs and application areas are not fertile ground for innovation. In most cases, this gives pretty straightforward indications for the rest of your digital transformation, your API program, and your plan for future hackathons.

However, it is more likely that a good idea came up, and won the hackathon. This section gives guidance on what to do next.

Assign a Dedicated Agile Team

Application leaders should meet with business sponsors after the hackathon to determine the next steps for bringing the winning hackathon proposals to market, and advancing their digital transformation.

They should select a subset of the proposals (there might be more than one winner) as potential products the company will commit to bringing to market, or at least to evaluate. Business leaders should select one or more proposals (usually no more than five or six to keep things manageable) that could have the biggest impact on the company's innovation strategy.

Assign a dedicated agile team that includes the winning hackathon team, IT and business (including a domain expert who will become the product owner) to each proposal (whether it is an individual or a team) that will be brought forward. Use lean startup principles to validate proposals and bring products to market.

Immediate follow-ups can include:

- Articulate and more precisely define user requirements and/or user stories for agile.

- Identify ways to refine the scope of the prototype (using techniques like lean startup or lean UX).

- Create a product plan and timeline, including minimum viable product release and sprint timelines.

- Identify milestones.

- Identify how the developer(s) will collaborate with the enterprise team (for example, will the enterprise provide office space, and when will the team meet again?).

Within this team, it is useful to assign a relationship and/or product manager to each participant proposal that will go to production, if the participant is interested and willing.

Build, Refine and Test the Solution

The objective at this stage is for the joint agile product team (the hackathon team, the business and IT) to bring the proposal to fruition as quickly as possible — very frequently as part of established products and solutions. This should all happen while ensuring that the product meets enterprise standards and governance requirements.

Potential steps to do this include:

- Refine the design of the app or solution — design the app and the user experience.

- Build — develop/construct the solution, beyond the initial prototype.

- Constantly refine — refine the solution based on user needs.

- Test — ensure the solution is ethical, reliable, scalable, secure and compliant.

- Beta release — test the solution in a relatively small and controlled environment, but with real-world usage.

- Production release — full production release to the target user base.

The cycles above might be interrupted at any stage, should the product team reach the conclusion that, for whatever reason, the product is no longer viable. Metrics to monitor the performance of each solution relative to the business sponsor's defined objectives are very useful. Measure these metrics to identify whether the hackathon achieved the objectives defined at the beginning.

Go to Market

The ultimate goal of the hackathon process is to put the new solution into production. Frequently, this is as part of an established offering, or implementing a new digital user interaction, through communications, pricing, sales engagement, etc. As an application leader, you will have a light involvement in the details of this — but you should still stay involved, and monitor how the solution or product is doing. You will learn a lot of lessons, which you can apply to the next hackathon and onwards. The metrics used in the previous section will still be very useful here, and will give true indications of effectiveness.

Plan for the Next Hackathon

As noted above, Gartner recommends that application leaders run hackathons on a consistent basis to accelerate digital transformation and continue creating an active community. The first hackathon will present evidence of the value that these events can deliver. It will show what works and what doesn't. Just as importantly, the lessons learned from the first hackathon can be applied to the ones that follow, and might possibly lead to targeting different developers, or opening up different datasets of application areas.

Hackathons are a journey, not a single event.

Very frequently, after a hackathon, the cost associated with it will be clear, and everybody in the company will look at the figure and ask, "was it worth it?"

Expect plenty of naysayers. It will take two to three *successful* hackathons to counteract those negative arguments, and there will always be some detractors. The point is that you have to associate the hackathon's cost with the *value* the company received from it. The value needs to be articulated clearly within your company in terms of the main points listed in the Things to Always Keep in Mind section at the beginning of this document.

Sometimes it will be possible to associate hard cash with the value, which can be offset from the hackathon costs. We strongly recommend to do that *only* when the amounts are credible and undisputable. Always be very conservative in estimating costs. A credible and undisputable low estimate makes a much stronger argument than an inflated but debatable high estimate.

The focus, however, should not be the financial expense. Costs are part of a much bigger picture of value, innovation, digital advancement, brand image and much more. The focus should be the frequently nonmonetary value that your company got out of the hackathon — or the lack of it because some hackathons don't work. When that is the case, recognize it openly and identify why. Point out clearly that the lessons you have learned from failure also have high value.

After all, the articulation of the value points above, whether positive or negative, will be your best planning tool for the next hackathon. Maybe you need different APIs, or a different group of participants, or a sharper focus on a different application area.

That's why the most important thing about hackathons is… *what you do when they are finished.*

Gartner Recommended Reading

Some documents may not be available as part of your current Gartner subscription.

"Adopting the Spotify Model for Better Enterprise Agile Scaling"

Evidence

[1] "Stay Focused and Keep Hacking," Facebook Engineering, May 2012.

[2] "Hacking at Employment Risks in Hackathons: Practice Insights," Seyfarth Shaw, Employment law Lookout.

[3] Techquisitions are acquisitions of digital or IT companies by enterprises in other conventional industries that have not created or sold information and communication technology-based products or services before.

[4] Acqui-hiring is the process of acquiring a company to recruit its staff rather than to exploit its products or intellectual property. Applies particularly to tech startups.

[5] "The Citizens Challenge," Citizens Bank.

More on This Topic

This is part of an in-depth collection of research. See the collection:

- Getting to the Details of the Digital Platform: A Gartner Theme Insight Report

GARTNER HEADQUARTERS

Corporate Headquarters
56 Top Gallant Road
Stamford, CT 06902-7700
USA
+1 203 964 0096

Regional Headquarters
AUSTRALIA
BRAZIL
JAPAN
UNITED KINGDOM

For a complete list of worldwide locations,
visit http://www.gartner.com/technology/about.jsp