



Academic Entrepreneurship for Medical and Health Scientists

Volume 1
Issue 4 *Ideation*

Article 2

9-24-2019

Rapid Prototyping Strategies

Olivia Jew

Perelman School of Medicine, University of Pennsylvania; The Wharton School, University of Pennsylvania

Roy Rosin

Penn Medicine

Follow this and additional works at: <https://repository.upenn.edu/ace>

 Part of the [Entrepreneurial and Small Business Operations Commons](#)

Recommended Citation

Jew, Olivia and Rosin, Roy (2019) "Rapid Prototyping Strategies," *Academic Entrepreneurship for Medical and Health Scientists*: Vol. 1 : Iss. 4 , Article 2.

Available at: <https://repository.upenn.edu/ace/vol1/iss4/2>

This paper is posted at ScholarlyCommons. <https://repository.upenn.edu/ace/vol1/iss4/2>
For more information, please contact repository@pobox.upenn.edu.

The *Academic Entrepreneurship for Medical and Health Scientists* book project is free to all – we don't ask for money but we truly value your feedback.

Below are two links -- one to a brief feedback survey and the other to a place where you can sign up to join our community of innovators and problem solvers. You can visit them and give tell us what you think now OR after you've had the chance to read this chapter -- either one works for us!

Please complete our brief feedback survey

<https://redcap.chop.edu/surveys/?s=HDXK3CE48L>

Join our growing community of Academic Entrepreneurs!

<https://bit.ly/3bnWTuD>

Rapid Prototyping Strategies

Summary

- Rapid prototyping allows a startup to test various versions or models of its idea in short iteration cycles.
- The intent of rapid prototyping is to learn from each iteration and avoid expensive mistakes that can result from untested assumptions.
- There are three main categories for rapid prototyping, and each category represents an essential question critical to success. First, does anyone want what you plan to do? Second, how will they interact with it? Finally, will it achieve a meaningful impact, moving the needle that customers want to see move?
- These three key questions map to three types of strategies: vapor tests, fake front-ends, and fake back-ends.

Creative Commons License



This work is licensed under a [Creative Commons Attribution-Noncommercial-No Derivative Works 4.0 License](https://creativecommons.org/licenses/by-nc-nd/4.0/).

Rapid Prototyping Strategies

Olivia Jew¹ and Roy Rosin, MBA²



Topic Relevance by Timeline

Summary

- Rapid prototyping allows a startup to test various versions or models of its idea in short iteration cycles.
- The intent of rapid prototyping is to learn from each iteration and avoid expensive mistakes that can result from untested assumptions.
- There are three main categories for rapid prototyping, and each category represents an essential question critical to success. First, does anyone want what you plan to do? Second, how will they interact with it? Finally, will it achieve a meaningful impact, moving the needle that customers want to see move?
- These three key questions map to three types of strategies: vapor tests, fake front-ends, and fake back-ends.

Introduction

In this chapter, we explore several strategies that can be used to implement rapid prototyping. These techniques can help propel an idea through the cycle of testing a completed product, learning from the results, and iterating. While these rapid prototyping ideas are most relevant for digital health technologies that lend themselves to short iteration cycles, variations on these themes can also be used with new healthcare services and medical devices, especially in the early stages of development or in preclinical testing.

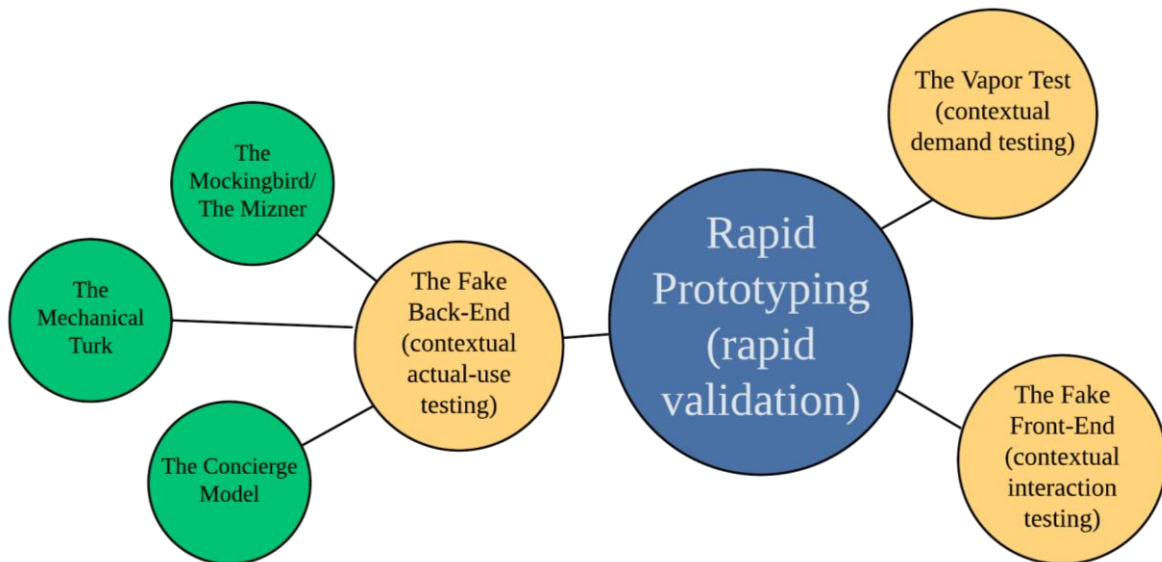
¹ Perelman School of Medicine, University of Pennsylvania; The Wharton School, University of Pennsylvania

² Penn Medicine

What Is Rapid Prototyping?

Rapid prototyping, also known as rapid validation when approached intentionally with explicit hypotheses, refers to the process of testing an idea as quickly and inexpensively as possible. The core concept is that it is preferable to find out early and at low cost that the team is heading in the wrong direction, rather than failing after already investing time and money into perfecting an undesirable product or service (see the chapter “Identifying Unmet Needs: Problems That Need Solutions”). When leading innovators say, “create a culture that embraces and celebrates failure,” what they really mean is to embrace fast, cheap failures methodically guided by intentional experimentation. Well-constructed hypotheses, ordered by what is most critical to success and least known or understood, identify the big assumptions that must be true in order to achieve a desired outcome. Experiments, dramatically accelerated by the methods we will discuss in this chapter, can then be designed and implemented to test whether one is heading in the right direction. This way, learning what does or does not work in days or weeks instead of months or years is efficient hypothesis invalidation rather than failure. By testing an idea as quickly and as cheaply as possible, the team will gain insights on how to improve the idea, and they will have more time and resources left to make necessary improvements based on believable evidence generated in a realistic context. Even if an idea is unpolished or underdeveloped, rapid prototyping can be used early in the process to determine what aspects of the product work, what assumptions about the product hold up, and what consumer interest in the product exists (see the chapter “Conducting Insightful Market Research”). This process is advantageous compared to testing a product after a large amount of time and money have been spent, as it may turn out there is no consumer demand or interest for the product, and all that time and money have been wasted. Validated data from the fast, cheap tests of rapid prototyping provide knowledge a team can use to reassess and build a product better suited for their desired end goal (Graham; Ries; “Prototyping.org”).

A quick internet search for “rapid validation” yields hundreds of different techniques, which all appear to be completely different—there are A/B tests, the Wizard of Oz experiment, crowdfunding, the concierge minimum viable product (MVP), landing pages, paper prototypes, or digital prototypes, just to name a few (Bank). However, we believe that at their core all these techniques can be categorized into three main buckets: the vapor test, the fake front-end, and the fake back-end (Figure 1). In order, these test whether anyone wants what the entrepreneur plans to build, how people will use it, and whether it achieves the desired results.

Figure 1. Categories of Rapid Prototyping Techniques.

The Vapor Test

The vapor test, also known as contextual demand testing or smoke testing, generates and disseminates a perceived existence of a product in order to test contextually whether or not consumers would be interested if the product actually existed. Asking people what they will do generates false signals; what people say they will do (e.g., buy a product) is fundamentally different from what they actually do, and the vapor test recognizes this discrepancy. Skilled researchers often say one must be careful to observe behavior instead of stated behavior. Stated behavior is what a prospective customer says they want or need, or it expresses the action they would theoretically take. Traditional research, including surveys, focus groups and interviews, falls into the trap of capturing this stated behavior (see the chapter “Human-Centered Design: Understanding Customers’ Needs through Discovery and Interviewing”). Prospective customers tend not to say what would accurately predict demand; this could be because people want to tell you what they think you want to hear, or they describe the way things are supposed to work instead of how they actually do work (e.g., men rarely tell you cutting their face with a razor is part of their shaving process), or they cannot imagine what they would actually do when presented with a new opportunity. For example, before investing in designing, building, and distributing a new product, a company might create a realistic digital representation of a product on an e-commerce site; should a prospective customer place the item in their shopping cart, the company might display an “out of stock” message as a soft landing. Prospective customers attempting to buy in a realistic context represents observable behavior and a strong signal of demand. This contextual demand testing works for new services as well. Leveraging a concept like a private beta, a company can describe a new service and add a button for signing up. Clicking the button may direct users to a page notifying them that the service is in a private beta-testing mode and is not accepting new clients at this time, possibly with the

option to sign up for the waiting list. While in some cases a private beta may simply mean that there are enough people already testing the product or service, in other cases it might mean the service or product does not yet exist; however, people signing up provides clear, observable behavior revealing demand. In the context of adding new features or functionality, or considering whether to build a new online service, this approach is sometimes referred to as a fake door approach, as named by Jess Lee, cofounder and CEO of Polyvore. Before investing the time to build a new feature, a site might add a link to the proposed functionality. Upon clicking and walking through that “fake door,” a user might see a message that it is under construction, as it does not yet exist. Yet the company now has a sense of interest. If there is limited interest below a cost-effective threshold, the company can decide not to move forward with creating this feature and save both time and money. This technique is best utilized when one wants to test whether or not demand for a novel product or service concept exists.

Indiegogo and Kickstarter are fully transparent versions of this concept, where one can test contextual demand by asking people to translate interest into an action like pulling out their credit card to reserve a product. In these cases, the products clearly do not yet exist, but the prospective customer’s action of joining the campaign to get it built becomes believable, realistic evidence of an unmet need.

The Fake Front-End

Sometimes the biggest unknown yet critical assumption is what a prospective user would do with a new product or service. The fake front-end, also known as contextual interaction testing, is used to test how someone will interact with an innovation that has been imagined and planned. A classic example, told by Alberto Savoia in his *Pretotyping Manifesto*, describes Jeff Hawkins’s prototype of the PalmPilot (“Pretotyping.org”). To test not only whether he might actually use a mobile device, but also how to design such a device so that it would be most useful in solving real needs that emerged in daily life, Hawkins fashioned a fake version of the device. This version was a block of wood roughly the size and shape of what would become the PalmPilot, with a simulated interface and a stylus also fashioned out of wood. What might one learn by carrying a block of wood around in their pocket? First, whether a user ever took it out of their pocket and wished it were real. Second, how to build it to most efficiently address the reason a user took it out of their pocket. If Hawkins wanted to look up a phone number or record an appointment, he would take out the woodblock as if it were a real, functioning mobile device and walk through the workflow using his imaginary product. From this exercise, he learned what features he found most useful and what designs would minimize effort while delivering the desired benefit, and he avoided investing time and money in building elements that failed to address contextual needs.

In the healthcare field, the Children’s Hospital of Philadelphia (CHOP) recently executed an exemplary fake front-end using this rapid validation approach. They examined whether certain children with sickle cell anemia (SCA) who presented at the hospital with fever could be sent home

instead of hospitalized. Historically, children with SCA who presented to the hospital with an elevated temperature were admitted to the hospital due to concerns including the risk of a serious bacterial infection (SBI). Some clinicians believed they could set criteria—proposing a potential algorithm—that would identify which children could safely be sent home instead of being admitted into the hospital. How were they able to safely separate the two groups, and, just as importantly, build support and buy-in while appropriately managing risks to move this potential breakthrough in care forward? Those at CHOP who believed they could identify which children could safely be sent home created a fake front-end algorithm using the criteria they believed would do the sorting. They began to assess patients with SCA who showed up to the hospital. Applying the proposed criteria, they determined whether to admit the child or not. But it was just a fake front-end, so, just like the block of wood, it didn't change what actually happened in the real world. All children were still admitted, but now there was an explicit record of who was identified as safe to send home, and the team could follow up to see whether it was the right decision or not. The system notified the Hematology Department about all low-risk SCA patients prior to their discharge from the emergency department in order to ensure follow-up within 24 hours. They were then able to inexpensively and safely evaluate whether this basic system accurately identified the low-risk SCA patients. Insights based on this simulation enabled iterations that refined the algorithm. Once they generated evidence that the system worked effectively as planned, they were then able to deploy a live version and make real decisions that prevented unnecessary admissions to the hospital. This approach saved the team from implementing an expensive or complex clinical workflow, launching a risky or misguided solution, or remaining stuck in old workflows. Today, more than a third of patients are sent directly home, avoiding expensive, inconvenient hospitalizations that could result in iatrogenic complications.

The Fake Back-End

A final essential question is whether the proposed solution achieves a materially better outcome. In this case, unlike a fake front-end, it actually needs to *do* something. Fake back-ends provide the mechanism for building something that actually works while staying true to the mantra of testing quickly at low cost. A central tenet of this method is avoiding the notion of building for scale (i.e., producing a solution that can handle high volumes and large populations) right out of the gate. Sometimes this is described as handcrafting an experience or building a “product” held together by tape, paper clips, and chewing gum that might work for three customers over two days but that could never scale, as it would fall apart under higher volume. This method shifts the focus from scale—usually a premature concern that could lead to scaling the wrong product—to getting it right and then scaling what works.

Some of the most successful startups in history took full advantage of the fake back-end. When Zappos started, nobody believed shoes could be sold online, as customers had to try them on before buying. To quickly test that assumption at low cost, Nick Swinmurn, the Zappos founder, started

selling shoes without having any shoes to sell. How? He went to a local shoe store with the proposition that he would take pictures of the shoes, then post them online. If an online customer then placed an order, Swinmurn would promptly go to the local shoe store and buy the shoe at full price, then ship it directly to the customer (Ries). He used someone else's shoes, at an unsustainable cost structure and level of effort (one that required real estate, inventory, and a lot of manual effort, all of which he could eventually eliminate) as a fake back-end, and it worked brilliantly.

Many healthcare breakthroughs share a common initial reality where building the ideal solution would take material time and investment, but generating evidence that raises the chances for securing investment and driving a strong return can be accomplished with a fake back-end. For example, the Hospital of the University of Pennsylvania (HUP) wanted to improve how they cared for women at risk of postpartum preeclampsia, which was the leading cause of morbidity and readmissions among this maternal population. A new standard requiring two blood pressure readings after discharge was known to keep these women safe. Despite best efforts and several attempts, HUP and other leading systems had not been able to acquire those two readings for a single patient. After setting up free walk-in clinics, flexible scheduling, and follow-up phone calls, success rate was still 0%.

Observations in clinics revealed that these younger women clearly preferred texting as a communication modality. This led to the assumption that sending at-risk women home from the hospital with a blood pressure cuff and texting them to acquire the blood pressure readings might work. Normally, you might build an automated system to execute this intervention, but the team recognized they did not yet know what to build in order to achieve high response rates from the discharged patients. A medical fellow pretended to be the system they might ultimately build, manually texting with the patients. Like the Wizard of Oz behind the curtain, this person was the fake back-end. Before investing resources to build an automated system, HUP could then test whether the system worked by implementing this rudimentary but functional small-scale version. Since they had a human, manual back-end, the team could rapidly test new approaches to elicit responses from women, iterating daily if needed. They cycled through personalization, various message timings, social support, and more before identifying a design that drove high response rates. In his *Pretotyping Manifesto*, Alberto Savoia refers to this type of fake back-end as a "Mechanical Turk," based on the eighteenth-century "machine" that seemed to have the ability to play chess, when in fact there was a small person with chess skills inside the box calling the shots ("Pretotyping.org").

Once the HUP team knew what to build, they transitioned to an automated back-end capable of scaling the intervention. Ultimately, with further work on patient identification, patient engagement, and care team response, they created a service called Heart Safe Motherhood that increased the success rate from 0% to over 80% and reduced morbidity and readmissions in this population by over 80%. This dramatic success was enabled by rapid validation, and the solution was scaled only after the team figured out what worked.

While fake back-ends are only temporary and only work on a small scale, they allow one to see what actually happens when people use the product. Fake back-ends come in a number of varieties and also enable testing with brief “mini-pilots” integrated into operations, to generate contextual evidence for whether to keep going, change direction, or stop. Two flavors of fake back-ends worth noting, in addition to the Mechanical Turk approach, are the concierge model and the mockingbird (Figure 2).

The concierge model involves becoming a person’s personal concierge and taking care of their every need. As someone’s butler, one can learn all of their preferences and constraints. Deep, contextual learning is enabled by walking alongside patients, clinicians, or caregivers for their entire journey, getting actively involved in addressing their needs. A team at Pennsylvania Hospital recently tried this method and discovered important, novel insights into patient populations they had served for years. They helped the patients they had adopted in the concierge model to get appointments, manage transportation issues, navigate medication complexity, problem-solve adherence challenges, and much more. Resulting insights revealed solution directions that had been overlooked. AirBnB’s early history contains great stories of leveraging the concierge model—for example, taking expert pictures of spaces for owners seeking to attract more travelers—to test hypotheses regarding what drove reservations. The insights from walking in another’s shoes and getting deeply embedded in struggles to accomplish tasks, both for insight and to build empathy, remain priceless (see the chapter on “Human-Centered Design: Understanding Customers’ Needs Through Discovery and Interviewing”).

The mockingbird—sometimes called the “Mizner,” after playwright Wilson Mizner, who once said, “If you steal from one author, it’s plagiarism; if you steal from many, it’s research”—is a fake back-end using a preexisting product close enough to an innovator’s new concept that it can be used for learning. This can be a difficult concept for innovators since their entire focus is doing something better than what came before, so why would one use what came before to test one’s idea? The reality is that a lot can be learned from watching prospective users try an existing, presumably suboptimal product to see how they use it, what users are seeking, and whether the existing product fails in the ways and for the reasons one believes. And since the product already exists, one can begin learning immediately and usually for a lot less money than when starting from scratch. If an academic entrepreneur has a novel idea for a task manager app because the thousand existing similar apps do not suffice, why not just put a dozen of those competitor apps in a dozen people’s hands to start learning, all for roughly \$12?

For instance, the Helen O. Dickens Center for Women’s Health at HUP wanted to reduce the burden of depression for antepartum and postpartum women. They hypothesized that an app combining regular, informational text messages and back-and-forth messaging capabilities could help mothers feel more emotionally supported by their provider. However, building such an app would have been a significant investment. The SPIRIT Group research team integrated Text-for-

Baby, a texting app developed by Johnson & Johnson, and MyPennMedicine, Penn Medicine's preexisting platform for communication between patients and providers, to create a theoretical app called MyPregnancy (Interview with Mahraj, Katy). Unfortunately, use of the MyPregnancy app showed no change in communication between mothers and their providers. Furthermore, in follow-up interviews mothers reported that the informational texts did not affect the level of support and engagement they felt in-between doctor appointments. Many apps offering content for antepartum and postpartum depression exist already; using a mockingbird test, the research team discovered there was no use in recreating those in an integrated app. By testing their hypothesis using a preexisting product, they avoided wasting time and money in developing a new app.

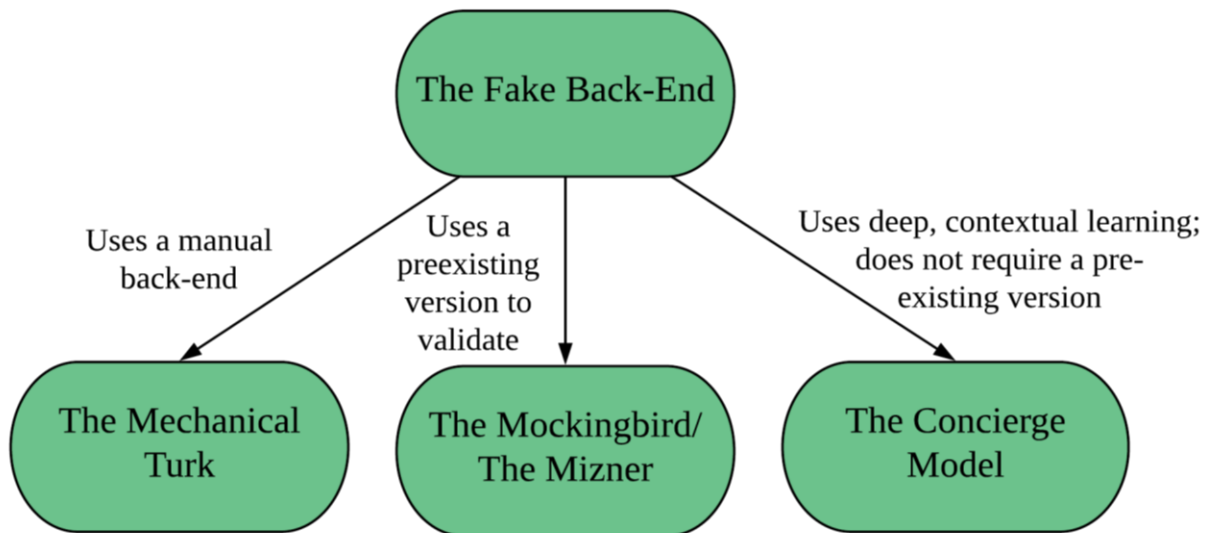
As made clear by the Heart Safe Motherhood story above, what fake back-ends allow is contextual testing: moving an idea into actual operations, even if only for a few patients, for a short period of time. We often refer to this operational method as a "mini-pilot," where one can measure actual impact in the context of a realistic workflow. This technique can be particularly useful when one is trying to validate an idea within a complex environment and the intent is to find out what kind of spillover effects the product might have. With a fake back-end, one can observe not only what happens as a result of the product but also the specifics of making the operation a reality. The Orthopedic Surgery Department at Penn ran such a mini-pilot to evaluate same-day scheduling, a simple concept that was not simple to operationalize and therefore required evidence to overcome inertia. The team constructed a creative fake back-end, in which they published the team lead's cell phone number on the website as the contact number for scheduling a same-day appointment. The team lead thus became a fake back-end call center, circumventing the entire machinery of Penn Medicine in order to quickly learn how same-day scheduling might work without prematurely changing core operations. During this pilot, patients could call a number and schedule an appointment for that day. While going through the steps of taking the call, scheduling the appointment, and seeing the patient, the Orthopedic Surgery Department learned how same-day scheduling could be operationalized once a system was automated and fully integrated. Furthermore, the department saw significant increases in conversion rate from patient interest to appointments and procedures, improved commercial mix, and a large percentage of patients who were not only new to Orthopedics but also new to Penn Medicine. With one physician willing to participate, they ran this mini-pilot for just a matter of days. The evidence they generated motivated change that led to the new same-day service being launched at scale.

Conclusion

Why is rapid prototyping important? As Harvard Business School professor Clayton Christensen said, "Statistically, 93% of all innovations that ultimately become successful started off in the wrong direction; the probability that you will get it right the first time out of the gate is very low" (Graham). Rapid prototyping will enable academic entrepreneurs to learn quickly at low cost, refining their offering to get it right before scaling. While perhaps initially discouraging, invalidating early hypotheses will impart invaluable insight regarding the target problem and the

proposed solution. A startup’s two limited resources—time and money—will also be put to use more efficiently. With these relatively small rapid prototyping experiments, the startup can test assumptions one by one, evaluate the idea piece by piece, and quickly amass a body of validated data with which the team can build and improve the product.

Figure 2. Relation Within the Fake Back-End.



References

- Bank, Christopher. “15 Ways to Test Your Minimum Viable Product.” *The Next Web*, 12 Nov. 2014, <https://thenextweb.com/dd/2014/11/12/15-ways-test-minimum-viable-product/>.
- Graham, Paul. “Do Things That Don’t Scale.” *Paul Graham*, July 2013, <http://paulgraham.com/ds.html>.
- Interview with Mahraj, Katy. 16 Aug. 2016.
- “Pretotyping.org.” *Pretotyping.org*, <https://www.pretotyping.org/>. Accessed 7 Sept. 2019.
- Ries, Eric. *The Lean Startup: How Today’s Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Random House Audio, 2011, <https://www.amazon.com/Lean-Startup-Entrepreneurs-Continuous-Innovation/dp/B005MM7HY8>.

Chapter Last Updated 9/24/2019.

Please check Scholarly Commons (<https://repository.upenn.edu/ace/>) for the most recent version.

The contents of this chapter represent the opinions of the chapter authors and editors. The contents should not be construed as legal advice. The contents do not necessarily represent the official views

of any affiliated organizations, partner organizations, or sponsors. For programs or organizations mentioned in this chapter, the authors encourage the reader to directly contact the relevant organization for additional information.

Content in this chapter is licensed by the editors under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) license.