The Scientific Method for Product Innovation
Lessons From Ten Years of Software Design

Henry Ford once said, “Anyone who stops learning is old, whether at twenty or eighty. Anyone who keeps learning stays young.”

I’m happy to say that in the ten years since we first opened our doors, we have managed to stay young, at least by Mr. Ford’s standards, because we have committed to never stop learning. In our quest to deliver a great user experience and to build new software products, we have tested our hypotheses by running experiments and made countless adjustments in the process.

Our learning has come from the hard work of the Digital Scientists team, a group of individuals who have demonstrated a passion for serving users by building purposeful new solutions. It also comes from our partnerships with dozens of forward-thinking clients, who have trusted us with their software products, their brands and their customers.

We offer our sincerest gratitude to everyone in that group. The last ten years would not have happened without you.

The greatest thing you can do with the knowledge and understanding is to share it. In that spirit, this eBook is our humble attempt at sharing some of our learnings from the last decade. We hope some of the lessons here will guide you on your journey toward building digital products that not only accomplish your goals, but also serve the needs of your customers.

Best,

Bob Klein
Chief Executive Officer
Digital technologies are disrupting whole industries and radically changing the way we do business. Companies that harness these digital technologies to deliver innovative products are better positioned to achieve a competitive advantage in the market.

Yet, the road to successful product delivery can be paved with near-misses, dead-ends and failures. Only 30 percent of executives at the world’s 2,000 largest public companies are “very satisfied” with their performance in converting ideas into market-ready products. And nearly the same number (28 percent) cite lateness to market as the key reason for product innovation failure.

Why does product innovation often fail or fall short? As with any project, the causes sometimes can be traced back to one of three common factors: Lack of time, lack of budget or lack of resources. In other cases, projects may lack stakeholder agreement, suffer from scope creep or get sidetracked as in-house IT staff struggle to make progress in the face of competing operational demands from the business. All of these can play a role in the success or failure of a digital product initiative. However, it is often the process — the way we go about digital product innovation and development — that is chiefly to blame. Successful digital product innovation requires radically different mindsets and methods than traditional software development.

In our experience, the four major keys to successful digital product innovation are: Think Big, Start Small, Move Fast and Iterate Rapidly. Companies that have adopted these proven approaches are typically rewarded with faster time-to-market, better scalability, lower development costs, enhanced user satisfaction and improved performance from their digital products.
Think Big

What Are We Trying to Accomplish?
Established companies may have made significant investments in building innovation centers that generate many interesting ideas, many of which could even result in products. Startups often take a “just do it” approach and start down the road of product development, failing to put the proper process and digital product development methodologies in place.

In many cases, businesses can spend years developing and perfecting these products without spending the necessary time on the front end showing the product to prospective users, determining whether the product fills a real customer need. By the time they uncover critical usability issues or product/market fit problems, it may be too late – too much money has already been spent – and the initiative or the startup fails.

As “The Lean Startup” methodology directs us, “The question is not, ‘Can this product be built?’ Instead, the question is, ‘Should this product be built?’ and ‘Can we build a sustainable business around this set of products and services?’” Those are the guiding questions that must drive our efforts.

To think big, we must start by having a clear picture of what overall success looks like. What is the business model and the key performance indicators (KPIs) for the new product (such as net revenue, usage or customer loyalty metrics)?

However, that is not enough. Before we go very far with the product idea, we must identify and validate user needs on the front end. We must learn what the customer problem is that we’re trying to solve. When enterprises spend sufficient time at the outset to conduct market research and analyze existing data such as customer satisfaction statistics, it can help ensure they are armed with the information and insights they need before embarking on a new digital product development initiative.

Start Small

Keep it Simple
Part of keeping things small is building the right core team made up of in-house subject matter experts as well as external digital innovation consultants. A small nimble team can often accomplish much more work and make more progress in a shorter period of time – with fewer communication snafus and other inefficiencies that may occur among larger digital development teams.

Starting out small doesn’t just make the project easier or more manageable. By working to solve a smaller business problem first, you can rapidly create a prototype or Minimum Viable Product (MVP) in two to three months or less. The advantage of deploying an MVP as a first step in the software innovation process lies in the build-measure-learn-feedback loop that helps you develop better products faster and more effectively – with better results.

In 2015, in partnership with Huddle, Inc., we launched a mobile ticketing solution for high school events. GoFan is a complete K-12 ticketing-platform, created in just 75 days. Fans can buy tickets online and from mobile devices. Tickets can be printed in advance or presented on a user’s mobile device for admittance into the game. Part of what made this multi-screen experience possible was starting with mobile first, and expanding to cover additional use cases, including: desktop, mobile web, iOS, Android, Gate validators, and Event admins.

Case Study: GoFan is a complete K-12 ticketing-platform that Digital Scientists created in 75 days.
Initially, we built a thin backend, then expanded it after proving the concept – GoFan now handles event ticketing, purchasing, and reporting for both printed and digital tickets.

**Move Fast**

**Don’t Get Bogged Down By Existing Processes**

The whole focus of a startup – or an innovation center within an established firm – is to transform ideas into products. Then, it’s critical to measure customer satisfaction and determine whether to “pivot or persevere” and accelerating the feedback loop through effective processes.

Markets move swiftly. If you wait a year, a competitor or startup could beat you to market. Since speed is critically important to product innovation success, it’s important that development teams avoid getting bogged down by existing processes, technology stack and tools on the front end. If these factors are allowed to dominate early efforts, they can impede or even doom the digital product innovation process.

In the beginning, focus instead on the design of the product and its market fit rather than on the technology stack. Modern technology tools and frameworks with an agile development process should be adopted and deployed while continuing to iterate the product. For example, depending on the business need, we might deploy an agile technology stack such as the following: Ionic Framework for hybrid mobile application development, Elasticsearch for super-fast search, AngularJS for the progressive application, Firebase for a real-time database and Stripe for payment applications.

Design Sprints, a concept developed by IDEO and Google Ventures, can be an extremely useful, five-day process for answering critical business questions for a product through design, prototyping and testing ideas with customers. The sprint enables you to see a finished “product” and gather customer reactions before making costly research and development investments.

**Iterate Rapidly**

**Assess Product Market Fit**

While many people may think the project is complete when the Minimum Viable Product (MVP) launches, we believe that this is where some of the best innovation begins. When it comes to product innovation, great is never good enough; there’s always room for improvement.

The iterative process is focused on measuring and assessing product market fit and enhancement. According to Marc Andreessen, market fit is the only thing that should matter to a startup. The iterative process is where we road-test the product and assess its market fit according to activation, acquisition, retention, referral and revenue. At this stage, users truly become part of the team. Their engagement and satisfaction with the Minimum Viable Product (MVP) can be measured, and this feedback can help drive data-driven decisions to power a better user experience – and create a better product that more closely matches the market’s needs. By definition, innovation requires new insights and new ways of seeing.
In the late 1990s, WebVan was going to transform the grocery business through online ordering and home delivery. But just a few years after being founded (and $1 billion in infrastructure investments) the company was bankrupt.

What happened?

A lot of mistakes were made. The case could be made that WebVan was simply too far ahead of its time. Given the popularity of online grocery ordering, pickup and even delivery today, it seems likely that WebVan paid a penalty for being too early.

But their biggest mistake was that they fell in love with their idea. Then they worked hard to make that idea a reality. A lot of really smart people held meetings and set budgets and timelines and executed their plans. They spent a tremendous amount of time and resources to pushing the idea forward and into the marketplace.

But there was something they didn’t do. They didn’t take the time to understand the market and the customer problem they were trying to solve. They didn’t test their product and validate their idea. They failed to realize that the market wasn’t quite ready for online grocery shopping. They created a well-developed product that failed to create enough interest to make it commercially viable.

Simply, they didn’t find the Product/Market Fit.

A Guide to Finding Product/Market Fit

The smartphone has spurred a digital revolution of which we’re still only seeing the beginning. Mobile technology has made it possible for users to be able to consume media, share content, interact with others, and purchase nearly anything from a convenient, handheld device. This has given rise to companies like Uber, Airbnb, and many others who continually raise the bar for functionality and user experience.
What is the Product/Market Fit?
In its time, WebVan was a cool idea. At the time, internet retailing was beginning to take shape and WebVan wanted to do for groceries what Amazon was doing for books. Since people bought groceries far more than books, it had the potential to save the average person hours of effort every week. WebVan was built on its coolness. They wanted to find out if they could build a new business model. They never asked if they should.

When it comes down to it, people don’t care about coolness. They care about their needs and their problems. If a product or a company can solve their problems in a compelling way that they’re willing to pay for, that’s what’s cool.

That, in essence, is what Product/Market Fit is: Understanding the customer problem that needs to be solved, understanding the marketplace and people’s willingness to pay for a solution, and testing the product to find the right approach that can then be scaled. You know when you have Product/Market Fit when you start to see user adoption.

Maybe they would have uncovered some aspect of grocery shopping that annoyed shoppers or identified a certain segment of the market that was open to their service. Then they could have built their business accordingly.

It’s easy to look at the Dark Ages of 20 years ago and marvel at how foolish we all were. But companies of today are not immune to the same mistakes WebVan made.

Entrepreneurs and Fortune 500 companies alike are bent on becoming the next Uber, disrupting industries or creating new ones. With so much energy, creativity and desire in the marketplace, along with consumer expectations on the rise, there is a great deal at stake. This has made the fundamentals — Product/Market Fit — more important than ever.

How to find your Product/Market Fit
Finding Product/Market Fit is the process of stepping back from your idea, understanding the customer problem you’re trying to solve, understanding the market’s desire for a solution, and testing prototypes to validate your ideas (or invalidate them and adjusting).

It’s simple in concept, but requires discipline. It seems countereintuitive at times, and requires a decidedly unconventional approach compared to what most executives are used to. But it’s the best way to arrive at a product that is truly innovative and will be embraced by the market. If nothing else, it helps you avoid investing huge sums of money in something that will never work.

In future articles, we’ll discuss specific methods for finding your Product/Market Fit. But here are four basic principles that can guide you as you embark on this journey:

Don’t trust your own ideas
At Digital Scientists, we don’t trust our own ideas, nor do we trust our clients’ ideas. Why? While some of us may be a good fit for the product (we love using apps as much as we love designing them), we’re too close to the product and too small as a group to give an accurate representation of the market. Customers, or the market, make the decision of whether your product will appeal to a large audience. And the market is always right. Instead of getting too enamored with your ideas, find out what the market wants. Find a problem that’s not being solved in a satisfactory way. Let the market pull the product out of you, instead of pushing it out.

A key part of this is understanding the size of the market. How many people are suffering from the problem you’re trying to solve? It may seem obvious, but it’s a key piece of information in building a sustainable business, and it’s often missed.

This requires looking at the big picture, taking several steps back and asking “why” repeatedly, until you find the root cause of the customer problem. It’s not easy, and it might require some outside perspective, but it’s the most important step.

It’s asking, “Should we build it instead of could we build it?”
Had WebVan determined their Product/Market Fit before they scaled their operation with massive infrastructure investments, they would have seen their planned approach was doomed to fail. And they also might have pivoted to an approach that would have worked.
Validate, learn, repeat

Absent a clear understanding of Product/Market Fit, companies—from the largest corporations to mom & pops—will go to a designer and ask for a web site or an application. The project will get whittled down based on what functions are possible in the given time and budget.

Eventually, the application will be built, on time and on budget, and it will be deemed a success. And it will still fail. This is what happened at WebVan, on a massive, billion-dollar scale. These conventions companies follow, of setting timelines and budgets and execution plans, present artificial constraints, and they have no place in the process of innovation.

It seems counterintuitive, but the process of finding the Product/Market Fit may require you to go over budget and past your timeline. It’s an iterative process characterized by creating, testing ideas, validating (or invalidating), learning, adjusting, and repeating.

That’s because you never know what the market will tell you. It might be trying to pull a product out of you that you never imagined. Putting a timeline on it will most likely cut it short before you arrive at the solution.

While the thought of going over budget may seem frightening, remember that it will still be a fraction of going all in to launch a product that’s fully developed, but not the right fit for the market.

Launch and test an MVP

The Minimum Viable Product, or MVP, is one of the best ways to develop and launch a prototype and find the Product/Market Fit. It involves creating a product in fairly rapid fashion, complete with bugs, glitches and other problems, and unleashing it on a test market to see if it works.

By determining your Product/Market fit, you know what your growth engine will be... who your customers are and what’s important to them...

That’s another frightening thing for many entrepreneurs and executives: the idea of putting out a product that doesn’t have all of the features and functions that were initially envisioned. But what the MVP does is gets the product in the hands of potential customers quickly. That, in turn, leads to faster feedback and allows you to make adjustments quickly, and based on real-user information.

That will give you the proof you need to determine what features users truly want and will use, and move on to the next phase of development.

Build, execute, scale...or abandon

Once you have proof that the product will work and understand how the market wants it, then your job is to execute—build, scale and market your product.

You can proceed with confidence, because by determining your Product/Market Fit, you know what your growth engine will be. You know who your customers are, what’s important to them, and how they are going to use your product.

You know this because you will have tested and adjusted, tested and adjusted, pivoted, adjusted, and tested again. You’ll know that you can and should continue to develop the product.

Or, you may not. You may not find the Product/Market Fit. The market may keep telling you, no matter what changes you make, that it doesn’t want that product. Certainly, that’s not what you want to hear, but it’s better than hearing the voice of a bankruptcy lawyer.

There are no guarantees, of course, but knowing you have found the Product/Market Fit gives you confidence. Confidence that you have the basis for a product that people will use and can be scaled, possibly to become a stand-alone business. Confidence from knowing your product enhances people’s lives through simplification, enjoyment, utility or purpose.

Confidence that comes from knowing you aren’t pursuing an untested idea, because the market told you it delivers real value.

The market may keep telling you, no matter what changes you make, that it doesn’t want that product.
Introduction

Right Size
Bigger is Not Better

If you’re a certain age, you might remember the childhood game of “telephone.” That quaint game of yesteryear also holds a lesson about teams for digital product development.

In the game, one child would whisper a message into the next one’s ear, and it would be relayed in the same fashion until it got to the last person in line. That child would then say the message aloud, and it invariably was different than the original.

The more children in line, the more the message changed. Laughter would ensue as the group tried to figure out who made the mistake.

It’s no laughing matter, but the same dynamic can happen if a digital product development team is not formed properly. Not only can messages be misunderstood, but other inefficiencies can doom the project.

Having seen and been a part of countless digital product development teams in partnership with our clients, we have developed a philosophy of what the right team looks like.

CHAPTER TWO

“The Right-Sized Team

If you can’t feed a team with two pizzas it’s too large.”

Try saying the following sentence with a straight face: “That committee really worked efficiently and got things done fast!”

You can’t do it, can you? That’s because committees by nature are slow, inefficient, and ineffective. Yet too often, when faced with a problem, companies throw bodies at it, essentially forming committees rather than teams. We’ve seen teams (sorry, committees) that included as many as six developers, not to mention people from marketing and the business unit.

There are several quotes from influential business leaders about team size, but our favorite comes from Jeff Bezos, founder of Amazon:
While some of us may like pizza more than others, the idea that teams should be kept to a minimum headcount is valid for a few reasons:

- **Small teams communicate better**
  It’s simple math, but the fewer people there are on a team, the less likely there is to be any misunderstanding. Communication is critical to any endeavor involving more than one person, and the closer that number is to one, the easier it is to communicate clearly.

- **Small teams are more accountable**
  With large teams (committees), it’s easier and more likely for people to not pull their own weight, hiding behind the efforts of others without being detected. This not only wastes hours and slows progress, but it also can lead to resentment. That doesn’t happen on small teams. People naturally want to do more because they feel vital to the team and to the mission. Team members are more involved, accountable, and invested in the process, which leads to the third point.

- **Small teams are more nimble**
  A key part of the digital product development process is learning. As we’ve discussed previously, it doesn’t work when a product’s parameters are dictated from corporate and turned over to a group of developers to execute. For a new product development team, it has to be okay to experiment and fail, and to learn from those experiments. On small teams, the learning curve is accelerated, as members are able to more quickly see what works and what doesn’t, and pivot accordingly with little explanation (because communication is easier).

Finally, there is an intangible dynamic, an esprit de corps, a togetherness that is more common in small teams. It’s not a guarantee, but small teams are more likely to buy into the mission, support each other, and attack the project with more gusto.

### Right Roles

**Skillset and Mindset**

For large companies, identifying the right people to take on roles within a product development team may seem like the easiest part. But it’s more difficult than it seems, and there is often at least one glaring omission.

The usual process starts with a product or marketing manager having an idea for a digital product that fills a perceived customer need, enhances the sales process, or otherwise makes life easier for customers. They get the idea approved and funded by corporate, and a team of developers is pulled in from IT.

People who work in corporate IT certainly have a great deal of technical knowledge, but it often does not include the knowledge needed to build new platforms. For example, they may not have skills in the newer technologies and oftentimes haven’t worked side-by-side with user-experience designers.

The success of the project depends on many factors, perhaps none more important than the team. The most effective teams are usually small and nimble, with the right people filling a few vital roles, and include some external partners.

### Don’t trust your gut... it’s imperative that you get your product into the customer’s hands as soon as possible.

But the mindset is not. People who work in corporate IT are used to working within a certain structure and set of rules. They don’t speak the language of building digital products, and they will naturally limit the product based on that point-of-view.

This is not to say that a corporate IT person doesn’t have a role on the team. That knowledge is important. But there is another role that is crucial and is almost always missing: the user experience (UX) designer. The UX designer is the voice of the customer. While other members of the team focus on company capabilities, goals, and competitors, the UX designer looks at the project from the customer’s perspective.

She knows the customer needs, and how they are likely to use the product to solve their problems, must be understood in order for the product to be successful. She plays a crucial role in developing a product that gains user adoption and is ultimately successful.

In our experience, the UX designer is most often an external partner.

### Right Partner

**A Fresh Perspective**

Imagine you’re working on a project, trying to solve a particularly vexing problem. You’re heavily invested in the project, toiling on it, yet no solution is presenting itself. Your view is colored by what you think is possible and impossible, the politics of your organization, and other priorities that may be languishing on your desk.

In other words, you’re in the weeds. Then, in walks an outsider, unfettered by the day-to-day realities of your company, offering a fresh perspective. With just a quick overview of the problem, he sheds new light on it and helps you see it in a different way. Suddenly, a pathway to the solution is clear.

Coming from outside an organization, external partners just aren’t weighed down by corporate bureaucracies and politics. They have the freedom to think big, fail fast, learn and adjust quickly.

For most projects, external partners can be a vital part of the team.

Developing a digital project is a huge undertaking that is foreign to most corporations. It often represents a significant commitment of time and resources. The success of the project depends on many factors, perhaps none more important than the team. The most effective teams are usually small and nimble, with the right people filling a few vital roles, and include some external partners.

While there are other factors that affect a team’s dynamics, its makeup goes a long way toward determining how well they work together, efficiently and effectively, to accomplish the goal.
It hits you suddenly. Maybe when you’re in the shower, or just going to sleep, or driving to work. That moment of inspiration, when an idea for a digital product simply enters your mind. Everything seems to make sense. You have a vision in your mind that’s profoundly clear. You know in your gut it will enhance your business, make your customers’ lives easier, and if you play your cards right, it will scale up to be its own profit center.

We hate to be the ones to tell you this, but don’t trust your gut! There is too much you don’t know yet. Too many variables you haven’t considered or even identified lurk around every corner. Do you really know how customers will use your product, or even if they will use it? Does it solve a problem for them? Is it the right problem? Is it even a problem at all?

Don’t get us wrong. A gut instinct can be a great start to an idea. But without answering the above questions, and many others, your digital product could be doomed to fail before you launch it, but not before you commit untold resources to developing it.

There’s only one group that can give you definitive answers: Your customers. So it’s imperative that you get your product into their hands as soon as possible. One of the best ways to do that is to develop a Minimum Viable Product, or an MVP.

**Defining Your Minimum Viable Product**

A faster, cheaper way to validate an idea
We’ve written about the MVP before, and for good reason. It’s one of the most efficient means of obtaining user feedback crucial to validating an
idea for a digital product, and scaling it to meet user needs.

Companies often assume they need to spend months or even years thinking through every last detail of a product, build a full-featured version, and unveil it to the market. This is a lengthy and expensive process with uncertain results at best, as user acceptance is far from guaranteed.

By contrast, an MVP is a quick, inexpensive approach. Our typical MVP process takes 90 days, and the cost is a fraction of conventional, full-featured product development. We don’t build the “complete” product because it’s not necessary or even advisable. The features and functions that will resonate most with users are still unknown.

The MVP helps us discover what features are most useful, how customers actually use the product, and how likely they are to adopt it on a large scale investment. Success, of course, is never guaranteed.

An MVP can be the right-sized approach to giving you a better chance of turning your moment of inspiration, your gut instinct, into a scalable, sustainable, profitable product.

Reducing Risk

We get it. Building an MVP and putting in front of users their feedback feels like a risky proposition. Typically risk-averse, corporations may default to the familiar process of conducting extensive market research, followed by months of planning, design, and development, to create a product that ticks all the right boxes.

That process feels safer, but in fact it’s inherently far more risky. It commits a great deal of time and resources to developing a product that, in the end, may not resonate with customers. In the time it takes to develop that product, the market may shift and render the product obsolete, a competitor might be first to market, or technology may evolve to shorten the product’s life cycle.

By contrast, an MVP is a far safer bet. It offers a tool for learning what will make your product successful in a shorter period of time, at a much smaller investment. Success, of course, is never guaranteed.

But an MVP can be the right-sized approach to giving you a better chance of turning your moment of inspiration, your gut instinct, into a scalable, sustainable, profitable product.

Four Steps to a Successful MVP

1. Define

The purpose of an MVP is to test a hypothesis that a digital product presents a compelling solution to a customer problem, or fills an unmet market need. So the first order of business is to define exactly what that problem or need is. A clear understanding of the question at hand is crucial if we are to find the right answers.

Having defined the problem, we then work with clients to prioritize what features to include in the MVP. This requires some restraint, as we don’t need to identify every last feature the product will ultimately contain, just the ones that will have the most impact in the testing portion. At this point, we also define the product architecture so the MVP can be built quickly and scaled later. Using the right “innovation stack,” we can focus time and resources on developing core features and use existing technology for supporting functions.

2. Build

With the product defined, we can start building. Precisely what gets built can take many forms, depending on the target platform(s) and the user experience we’re trying to deliver. It’s here that the importance of following an agile process comes in. Learning is inherent to the MVP process, and when working with new technologies, unforeseen opportunities, ideas and obstacles will likely present themselves. As a development team, we need to be able to pivot, leveraging new information to incorporate it into the product.

Knowledge of solution architecture is also important in the build phase. As mentioned above, we identify the right innovation stack upon which to build the product early in the MVP process. This makes it easier, faster and cheaper to build standard functionality, such as messaging, mapping and search. Thus, we can then focus on the aspects of the experience that need to be unique or special.

3. Launch

This is where the real fun begins. Get the product out the door and let users have at it! Keep in mind it is important that before you do launch, identify the hypothesis of your MVP. What is it aiming to confirm or challenge? Once you determine this, it’s important get your MVP to market – getting as many users as possible to confirm or even deny the hypothesis.

Through the analytics and user behaviors you observe after the launch, we will obtain a great deal of information. The key to learning is to look deeply at the data. Not just surface-level data about if users are adopting the product, but how they are using it. We want to track individual events that show how people are moving through the product, and how that compares to expectations and assumptions.

It’s rare that anyone gets it right on the first try. That’s why the MVP, and our define, build, launch, learn approach to it, is designed to be an iterative process. Multiple iterations are crucial to finding the right formula for a product that users will embrace.

4. Learn
Many corporations simply aren’t structured correctly to create digital products that make the user the priority.

All too often, when a company is presented with the opportunity to develop a digital product, the default is to bring in the IT department and hand it over to them. This is understandable, because they are the closest people who have at least a working knowledge of software development. The result, however, is that product launches become IT projects.

This is not a good thing, because there are some fundamental differences between an IT project and a product launch. An IT project typically involves adding, enhancing or altering the functionality of technology that sits on an existing enterprise platform. The goal is simply to complete the project as requested and make it work within defined parameters.

### Managing an IT Project vs Launching a New Product

<table>
<thead>
<tr>
<th></th>
<th>IT Project</th>
<th>Product Launch</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mindset</td>
<td>Delivery - Focused</td>
<td>Learning - Focused</td>
</tr>
<tr>
<td>Success Metrics</td>
<td>Time, Budget, Requirements</td>
<td>User / Customer Adoption</td>
</tr>
<tr>
<td>Team Structure</td>
<td>Siloed</td>
<td>Collaborative</td>
</tr>
<tr>
<td>Process</td>
<td>Waterfall</td>
<td>Agile</td>
</tr>
<tr>
<td>Technology</td>
<td>Corporate Stack</td>
<td>Innovation Stack</td>
</tr>
</tbody>
</table>
A product launch is something different entirely. There are far more unknowns than knowns, chief among them is how users will adopt the product, if at all. Launching a product requires experimentation, iteration, learning and adjusting. These are fairly foreign concepts within corporate IT culture, which creates some daunting roadblocks that can easily prevent a product launch from being executed effectively. And may derail it entirely.

In our experience, there are three major roadblocks standing in the way of corporations launching an innovative product. And there is one way to start breaking down those roadblocks.

### Roadblock 1: Standard Operating Procedures

Corporations have processes. Lots of them. It’s a fact of life that won’t change anytime soon. And while they are necessary to keep large corporations operating smoothly and efficiently, they are not conducive to effective digital product launches.

Corporate IT departments follow procedures. They focus on getting the task done, according to the parameters given, the timeframe and the budget. Their job is to make something work according to the way they’re told. No more, no less.

They follow prescribed internal processes for scheduling a project, assigning a team, planning, development, corresponding with leadership, and producing a deliverable. They do what they’re asked.

If they don’t follow processes and procedures, their jobs are on the line. Their focus is on executing on a prescribed set of requirements, and they’re good at it. But “learning” and “discovering” are not in their job descriptions. There is no incentive to innovate, experiment, or fail — all of which are crucial to the product development process.

### Roadblock 2: Legacy Technology

Nearly every corporation in the world has two or three technology platforms — Microsoft, SAP, Oracle, and a few others — that they use to run the various aspects of their businesses. The people responsible for managing those platforms work in corporate IT.

So what happens when IT is tasked with building a new digital product? More often than not, they default to the legacy technology stack. The problem is, these legacy stacks are most often not the best options for new digital products. They may not deliver the right functionality, speed, or experience that will resonate with users. The most successful companies launching digital products today (e.g., Facebook, Airbnb, GE, Wal-Mart) have adopted newer, more open technology stacks. Here at Digital Scientists, we refer to this as the Innovation Stack.

Corporate IT departments, as noted above, are not incentivized to innovate. They’re risk-averse by nature, and they’re paid to make sure things work and they don’t break. And if they do break, they need to be the ones who fix them.

So they use the tools and technology they have at their disposal, defaulting to the same two or three legacy technology platforms. Focusing on function over form, they shoehorn the product into the technology platform, and that usually results in more time and higher costs. Worse, this approach doesn’t take the user needs into account.

### Roadblock 3: Internal Politics

This is the big one. For better or worse, politics are inextricably tied to corporate culture, no matter how progressive or forward-thinking a company may be. Politics vary from company to company, of course, so it’s impossible to generalize. However, when it comes to digital product development, politics play a big part in determining which projects get approved and funded. Typically, they’re the projects that address the most immediate needs, which means sales, cost reduction, or a combination of those.

This is a short-sighted approach, where internal problems and pet projects take precedence over solving customer problems. Even when customer problems are addressed, it is not a strategic process based on understanding what the problems are. Instead, it’s ready, fire, aim. The product is developed under an arbitrary schedule and budget, based on untested thinking.

When the political climate within a company is driven by short-term results, there is little room to test new ideas or explore future business models. No experimentation. No innovation. This can kill a digital product launch before it even gets to square one.

### The Solution: User-Centered Design

There is one thing companies — and people responsible for leading digital product launches — can do to start to break down these roadblocks.

Approach everything with a user-centered mindset.

The user experience you’re trying to deliver should drive everything: processes, technology, and even politics. Your focus should be on developing something that’s different and unique in the marketplace. If that means you use an unfamiliar technology or follow a new process, so be it.

The user’s needs are what they are. If they are presented with a product that doesn’t solve their problems, save time, or enhance their lives in some way, they can and will reject it.

It sounds simplistic, but taking a user-centered approach can help you go from derivation to innovation, evolution to revolution. Knowing what’s best for the user, and delivering it, is the only pathway to a successful digital product launch.

Typically, successfully funded projects address the most immediate needs—sales, cost reduction, or a combination of those.
When developing a digital product, speed is the priority, product/market fit is the goal. You need to get the product into users’ hands fast so you can begin the learning process and validate your idea.

The best way to do that is to launch a minimum viable product (MVP), and the innovation stack is the MVP’s best friend.

A digital product is made up of a collection of functions and features. A few of those features are core components of the product. They are central to what the product is — unique, proprietary elements that make the product special and compelling to users.

But some, in fact most, of the functions are not unique to a product. They are standard elements that support the key functions. They might include data collection, payment processing, messaging, and calendar interfaces.

So, when you’re developing your digital product, knowing you have to get your MVP launched quickly, and that you have finite resources, where do you want to focus your attention? Of course, you want to focus on the core functions of your product that will add value. But you still need the supporting elements.

The innovation stack helps you understand your product so you can use existing, “off-the-shelf” solutions for the supporting elements, saving time and money.

What is an Innovation Stack?
You may have heard the terms “technology stack,” or “software stack.” While these terms have the same basic meaning, we prefer “innovation stack,” because innovation is at the heart of digital product development.

An innovation stack is a group of programs, technologies, or applications that work in tandem to produce a result or achieve a goal. It’s called a stack, because the programs are literally stacked on top of one another, digitally speaking, in “layers.”

The innovation stack is different for every digital product, but the basic layers include:

• Data
• API
• Web Server
• Application
• Admin
Within these layers are the tools, software packages, and solutions, that work together to make your product achieve its intended goal. This includes the existing technologies that can be plugged into your product to support the core functions, as well as the code for the core functions themselves.

To understand how the layers of an innovation stack work together, let's use a simple example. Say your product is a web application that displays local weather forecasts. A user logs onto your application and wants to see a five day forecast for Atlanta, Georgia.

To get the information from the database to the user is not as simple as the application just grabbing it and displaying it. The information has to be processed, filtered and formatted so the user sees what she needs. Nothing more, nothing less.

### Open Adoption Software Drives Innovation

Understanding your product’s innovation stack is crucial to launching an MVP, because it helps you make decisions about where to focus your efforts and where you can leverage ready-made solutions. And there are many different solutions available thanks to the trend toward something being called “open adoption” software.

In contrast to proprietary licensed software, like Oracle and SAP, open adoption software can be used without the purchase of costly seat licenses while still being able to scale as required. The innovation stack is made up almost exclusively of open adoption software and SaaS services - facilitating ease of development and speed to market.

The open adoption software trend is driving innovation. With so much new software available, it’s enabling developers to launch MVPs quickly and for far less money. It allows you to move fast. Instead of having to spend time writing code for basic functions, you can simply plug them in. This means you can keep your team small, and it makes it easier to iterate, testing your product until you find the right approach.

Most of all, it allows you to zero in on what’s important. We’ve stated the importance of the user experience (UX) in determining what a digital product should be, and that same principle applies to the innovation stack. The user needs you’re trying to meet, and the experience you’re trying to deliver, should ultimately drive what’s included in your product’s innovation stack.

It should not be determined by what your corporate IT department is familiar with or comfortable using, and that leads us to the importance of why you should be aware of the innovation stack. When you’re building something, it’s important to know and understand what tools you have available, even if you’re not the one using them.

Knowing that you have access to virtually unlimited tools beyond your enterprise software system means that you can ask the right questions of your development team. Most importantly, it means you can confidently assemble and manage your development team and focus your resources in the right places.

So you can launch your product fast.

---

**An innovation stack is a group of programs, technologies, or applications that work in tandem to produce a result or achieve a goal.**

MVPs quickly and for far less money. It allows you to move fast. Instead of having to spend time writing code for basic functions, you can simply plug them in. This means you can keep your team small, and it makes it easier to iterate, testing your product until you find the right approach.

Most of all, it allows you to zero in on what’s important. We’ve stated the importance of the user experience (UX) in determining what a digital product should be, and that same principle applies to the innovation stack. The user needs you’re trying to meet, and the experience you’re trying to deliver, should ultimately drive what’s included in your product’s innovation stack.

It should not be determined by what your corporate IT department is familiar with or comfortable using, and that leads us to the importance of why you should be aware of the innovation stack. When you’re building something, it’s important to know and understand what tools you have available, even if you’re not the one using them.

Knowing that you have access to virtually unlimited tools beyond your enterprise software system means that you can ask the right questions of your development team. Most importantly, it means you can confidently assemble and manage your development team and focus your resources in the right places.

So you can launch your product fast.

---

**THE INNOVATION STACK**

The layers of the innovation stack all work together to support your new product. The solution architecture (or the specific technologies for each layer of the innovation stack) is typically driven by the requirements of the user experience. We have shown some examples in the layers below.

<table>
<thead>
<tr>
<th>Layer</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application Layer</td>
<td>What the user sees and interacts with. It’s made up of screens, menus, images, input fields, and other tools to enhance the user experience.</td>
</tr>
<tr>
<td>Data Layer</td>
<td>All of the information required to provide the user with the information she wants resides in the Data Layer.</td>
</tr>
<tr>
<td>Web Server</td>
<td>Used by the application administrators to manage all of the basic software configurations and functions of the application.</td>
</tr>
<tr>
<td>API Layer</td>
<td>Provides the information to the Application, which displays the information in a user-friendly, easy-to-understand way.</td>
</tr>
<tr>
<td>Administration Layer</td>
<td>Liaison between the back and the front of the application. It processes, filters and formats the information request and sends it down so the API can gather the information from the database and other sources.</td>
</tr>
</tbody>
</table>
Congratulations! You have a digital product. You’ve determined the right Product/Market Fit, you’ve assembled a lean, efficient development team, and you’ve launched a Minimum Viable Product (MVP) to validate your idea. Users are adopting your product and providing feedback.

Sure, you need to refine and polish it a bit, maybe add a feature here and there. But the product is basically done, and all you need to do is sit back and reap the rewards, right?

Once again, we hate to be the burster of bubbles, but you’re not done. When it comes to building an innovative product, you’re just getting started. In fact, you may never be done.

This is the point where many digital products and companies took a turn from what they initially had planned to the innovative, user-specific products we see today. Match.com, for example, started life as an online classified advertising platform, and now is one of the top online dating services.
Chapter 6

Fueling Innovation

What happened to Match.com? How did their product change so much? The answer, simply put, is that’s where their users took them. It sounds counterintuitive, but companies are not completely in charge of what their digital products ultimately become. The smart companies understand that a successful product launch results in the emergence of a user group, whom companies have a vested interest in serving by offering new, innovative experiences.

To fuel that innovation, companies should follow a user-driven roadmap that ultimately leads them to create a product that meets a specific need and enhances their customers’ lives in a compelling way.

As its name suggests, a user-driven roadmap is a strategic, high-level document that captures the vision and objectives of the product, guiding its development and evolution over time. It serves as a reference for making decisions about what features to add to a product or how to change its functionality.

One of the primary sources of input for the user-driven roadmap is — you guessed it — users. Through their feedback and usage behavior, they provide valuable information that can be used to create and revise the roadmap.

The beginnings of the user-driven roadmap are actually put in place before the MVP is developed, with the business objectives, purpose and vision for the product as the first guideposts. Starting with some core functions, developers build an MVP as the means of generating user feedback. By getting the product in the hands of users as quickly as possible, developers essentially make them part of the product development team.

Once the MVP is launched, users begin providing feedback on the product through how they adopt the product and how they actually use it in real-life situations. This provides clues on how the product should evolve, what features should be added, and what’s most important to users.

As noted above, specific user groups may also emerge, gravitating to the product in greater numbers than other users. This is what happened to Match.com, when people began using what had been a classified advertising product for online dating. Serving this user group demanded that certain features and functions be added. Match had to respond, or risk losing the users they had cultivated.

Iterative Process

Iteration is central to the MVP and creating a user-driven roadmap. The MVP is an efficient means of testing a product and determining market fit based on user activation, acquisition, retention, referral, and revenue.

It is the process of taking the core components, testing them through the MVP, and iteratively adding features based on user feedback. With each iteration, the features are tested to determine whether they add the value, broaden utility, and generally make the product better or more useful to the user group. The features are changed, refined, added, or eliminated, and tested again.

The way people use products and the emergence of new user groups can happen quickly, or it can happen gradually over a long period of time. For that reason, the iterative process — and the learning that results from it — never stops.

Creating a Platform

Users are a key influence in creating the product’s roadmap, but they are not the only factor. As a company, you don’t want to simply do everything users seem to demand. You have your own business objectives to take into account. Your digital product has to support those business objectives in addition to serving the user groups.

That requires first having a clear vision of your business and the objectives you want the product to support, then prioritizing the user-driven changes you want to make. Those two axes, the user feedback and your business objectives, are what make up the roadmap that will guide you in the continual development process of the product.

Through this process, you’re evolving your product into a platform that can adjust to the changing needs of customers and the changing dynamics of your industry and business strategy. The technology will stay largely the same, but the way people use it will change over time.

Listening and Understanding

It’s an old saying, but developing a digital product is a journey, not a destination. Business objectives, market dynamics, and user needs are constantly changing at unpredictable rates. Maintaining user loyalty and adoption requires a commitment to continual innovation.

Every journey requires a map. A user-driven roadmap — based on obtaining and listening to user feedback — is a crucial element to making data-centered decisions to create increasingly better user experiences.

In addition to listening to user feedback, creating an effective roadmap also requires the ability to understand it. There is a difference between what users say they want and what they will actually use. The ability to read the subtext of user feedback and acting on that information can mean the difference between delivering a product that loses users, and one that delivers real value while meeting business objectives.


About Digital Scientists

We are your software innovation lab – an integrated team of product designers and software developers using the latest frameworks and tools to deliver awesome new software products. We partner with forward-thinking businesses to create mobile apps, websites, ecommerce stores and internet of things (IoT) applications. We believe in taking a “design thinking” approach to solving complex problems for users. Established in 2007, we are proud to call metro Atlanta home.

For more information visit digitalscientists.com