



What Can You Build?

Some ambitions are bigger than applications.
We asked successful devs turned founders:

“What makes a software developer
a great business developer?”

```
// Features
// Oct-Nov '16
```

```
Build Better;
```

```
name Jeff.Atwood
```

```
{
  developer class GrandMaster
  {
    career Options? _noLimit;
    career Path? _relativeToDrive;
    career TimeSpan? _noExpiration;
  {
    get
    {
      return _careerInsights;
    }
    set
    {
      if (value == iconStatus)
      {
        throw new IdeasSolutionsCareerPathsEtc.(
          nameof(IdeasSolutionsNow),
          value,
          "The value is self evident!");
      }
      _stackoverflowStackExchange = w00t!
    }
  }
}
```

```
/// <summary>
/// Jeff talks about his path to success from
/// A pre-internet desk job to helping devs
/// Build code and communities thru
/// Two preeminent "Stack" businesses
/// Now he's helping devs interact with humans!
/// </summary>
```

```
page Detected
```

```
{
```

```
get pageSeventeen;
```





The Code that Rocks the Cradle

This coding mom turned an over-stuffed garage into big business.



When Your Backup Plan Becomes Your Biggest Success

Developer turned author, turned founder shows us there's one simple key to success—whether you're building software, writing a book or steering a company.

3 Skills Devs Need NOW that They Didn't 5 Years Ago

Gone are the days of picking a task off a whiteboard and going into a hole for three days. Here are the skills you need NOW to be successful.



Publisher: Matt Watson, Stackify
Editor in Chief & Creative Director: Max Hobbs
Managing Editor: Jayme Thomason
Technology Editor: Taylor Ford
Business & Sales Editor: Michelle Harris
Art Director: Kevin McCoy
Production Manager: Jennilee Tangpuz
Marketing & Promotional Partner: Brink Insights

Key Takeaways from this Issue:

There's a point where you know you've accomplished the goal. That may be the time to start something new. Take what you've learned about your first company, the successes, the failures the unmet needs, and build on top of that to create success in your next venture. —*Jeff Atwood, p. 17*

If you want something done right, understand your own strengths and weaknesses. Know what parts you can do yourself, and know what parts will benefit from building a team with skills that complement your own. —*Steve McConnell, p.14*

Coding and building a product is not the only side of the business when you are a founder. Pay attention to marketing, sales, and aspects other than just the development. If you can't take it all on, build a team that will work well with you and support your goals. —*Alexis Pratsides, p.13*

Build your team up with people who share your passion for your project, and consider using interns for some early work. Be mindful of the legal world, but don't lose sight of what made you start the journey in the first place. —*Todd Moore, p.10*

Your next greatest success may come from simply solving a problem for yourself or friends. Your technical background is an asset, but don't forget to practice the art of good communication. —*Stephen Foster, p.21*

Your big idea may come from what you are most passionate about. Although, don't forget to focus on the business as a whole, and not just the development work. —*John Negrón, p.20*

You don't need to strike out on your own to launch the next great tech product. Sometimes, innovation starts within another organization and then branches out on its own. —*Robert Collazo, p.9*



For the Oct/Nov issue of BuildBetter Magazine, we conducted 13 interviews with founders and co-founders who began their careers as developers and made the leap to start their own companies. From my vantage point (i.e. the best seat in the house), the stories we heard across the complete developer to founder cycle and all its stages were amazing. We spoke with a few developers just starting their entrepreneurship journey, a few in growth mode, several leading the organizations they'd always dreamed about, and one who left to start it all again.

As I reflect on each of the stories, there are key learnings for all of us (founders, aspiring founders, and those who just want to code) because we have two things in common—**building things** and **solving problems**. These passions run through our blood whether we're building apps or building companies.

So why do some developers stay developers, and why do some take the leap to entrepreneurship? While each of the founders we talked to had different journeys, each came to a point where they saw a problem that needed solving and they had a decision to make—do I stick with my job or give it a go?

As we explore the stories of these founders, we've highlighted key learnings for you if you're on the **founder track** (highlighted in purple), and we've also highlighted quality insights for you as you develop in your **programming career** (highlighted in green). There's something for everyone and I hope you get as much out of this issue as I did.

Keep Building,

Jayme
BuildBetter / Managing Editor



| v1.3 | OCT-NOV 2016

Build/better

Devs & Founders

Some developers are great at building more than software.

Some developers build businesses that exceed everyone's expectations, including their own. Learn how the developers we interviewed in this issue did it!

- p.6 **Find bugs before they get hairy**
Matt Watson: Stackify
- p.9 **Fulfilling a need, founding a company**
Robert Collazo: HelpSocial
- p.10 **White noise & coding for success**
Todd Moore: TMSOFT
- p.13 **The twist at the crossroads**
Alexis Pratsides: MintTwist
- p.20 **Developing for change**
John Negron: Be a Doer
- p.21 **Coded to teach**
Stephen Foster: ThoughtSTEM

★ Don't miss the killer profiler promo:
Free tools and T's!



It's the bugs you can't see that get hairiest.

Matt Watson / Founder & CEO / Stackify

Our fridge at home is always full. This is generally a good thing. The downside is, it's hard to find stuff in a fridge that's been packed to survive an apocalypse.

When I can't find something, there's a chance someone finished it off. However, certain items are simply MIA; some sort of Bermuda Triangle phenomenon. In fact, on a recent snack-finding mission I finally found the left-over carbonara I'd been hunting for weeks. Unfortunately, it now looked more like an aged wookiee than a meal.

Isn't coding just that way? No one wants to deal with bugs, but if you can find them early on, you can deal with them before things get out of hand. The longer it takes to find them, the hairier and gnarlier things can get. The sad fact is, finding app issues is often harder than finding the grape jelly or the last deviled egg.

While there is no hope for my fridge (aka the "Upside Down"), there are some great tools available for finding nasty code issues before they sprout hair or spawn putrescent Gremlins or something.



It's the bugs you can't see that get hairiest, Cont.

Hooray for Profilers!

I love code profilers. I've dedicated the last several years of my career to code profiling technology. In my experience, there are three types of profilers:

1. High Level

A high level profiler tracks performance key methods in your code. These profilers typically handle transaction timing, such as tracking how long a web request takes, while also giving you visibility to errors and logs. This is generally what you want on a server to track the performance of your app when you have thousands of users logged in. Because they are very high level, they are very low overhead. These are generally known as APM products.

Specific products include:

- Retrace (Stackify)
- New Relic
- Application Insights (MSFT)

2. Low Level

Low level profiling usually tracks performance of every single method in your code and, potentially, every single line of code within each method. It can be very slow with a lot of overhead. These types of profilers are also tracking memory allocations and garbage collection to help with memory leaks. They're very good at finding that hot path, figuring out every single method that's being called, and what's using the most CPU.

Specific products include:

- Visual Studio Profiler
- ANTS (RedGate)
- dotTrace (JetBrains)

3. Hybrid

A hybrid allows you to get a deeper level of detail, but runs on your desktop so you can use it every day. These can track every single transaction that happens, including the performance of key methods, dependency calls, errors and logs. As with server-based APM tools, hybrid profilers provide a relatively high level overview with low overhead, but you can run them on your dev box to find and fix issues before you deploy.

Specific products include:

- Prefix (Stackify)
- Glimpse
- MiniProfiler

Naturally, I'm partial to Stackify's Retrace and Prefix tools. They've been a labor of love for the past four years.

It's the bugs you can't see that get hairiest, Cont.

Retrace (QA/Test & Production Tiers)

Retrace (formerly APM) provides server-based performance profiling for .NET & Java applications. Retrace helps dev managers, architects, and team members understand the performance of all apps and servers to quickly find and fix application problems.

With Retrace (starting at \$10/mo/server) you can view every detail of every transaction, including web service calls, application logs and errors, SQL queries, and dependencies.

Retrace is designed to give developer teams a leg up on apps once they've been deployed. View performance of all apps, monitor performance of key transactions, track server and application metrics, leverage detailed transaction traces, identify all application errors, and search across all application logs in one powerful tool.

Prefix (Free & Premium Licensing)

Prefix provides code profiling for .NET & Java applications on developer's desktops. Prefix is currently being used by thousands of developers in well over a hundred countries.

This lightweight profiler helps you debug and prefix application problems before they get to production. The free version gives you visibility to every single web request, SQL query, and web service call, as well as caching, queuing and lots more.

The premium version (\$15/mo) takes it to the next level for teams and power users with the ability to profile methods of your own code as well as non-web & self-hosted apps. You also get end-to-end transaction profiling, longer retention of trace data, the ability to save and share traces, and more—**including a free QA/Test license of Retrace!**

Profiler Promo!

Get the deal!



Use Promo Code
BB16OCT

Install & activate
Prefix Premium
or a **Retrace trial**
by **11-15-16** and
get some killer
code swag.



Often, the apps and products we create are born out of necessity, and founders are born from that simple need-based-solution. Robert Collazo, co-founder and CTO of [HelpSocial](#), found himself in this position while working as a system admin. Out of a need for social media management team-based software, HelpSocial was created as a bit of a happy accident.

They say that necessity is the mother of innovation, so innovation must then be the father of devs-turned-founders. While working for Rackspace, Collazo developed a program to improve communication between customers and Rackspace over social media. During conferences, they would display their tool on big screens, exposing his product to a mass audience. People quickly took notice of what a great tool Collazo had created, and client requests came pouring in. "We were Rackspace employees, but as soon as we raised enough money to go out on our own, that was it! We were no longer Rackspace employees, and Rackspace became a customer."

When you pass that line into founding a company and the product you are selling is owned by you, there are often unexpected obstacles in the way. From building a team, learning about marketing and sales, to legal concerns. "Working with lawyers is no fun. There's a lot of paperwork and working with investors. There's a lot to do to protect yourself, your company, IP, and more." Although, if you surround yourself with the right people, not just in terms of skills, but people who believe in you and your product, that can make a world of a difference. Collazo found that his team has been an invaluable asset. "When you are a startup, it is really important to have people who believe in what you are doing. We can't pay people those big salaries that allow them to jump from company to company, so it is really important to have people who are dedicated and want to work for you."

When you make that switch from being an employee to founding your own company, it becomes a lifestyle change. "Unless you've done it, you don't understand. I have a couple friends who own businesses; they're the ones that empathize the most with me now." Finding the right team members, securing yourself legally, and growing your passion into a successful business will no doubt be a satisfying endeavor. Whether by a happy accident or a lifelong pursuit, it all begins with an idea. ■

Fulfilling a Need, Founding a Company

Robert Collazo

Co-founder & CTO / HelpSocial

White Noise & Coding for Success

Todd Moore

*Developer of White Noise /
Founder of TMSOFT*

When you dream of founding your own company, you probably have a vision in your head of what it looks like. While everyone hopes that they launch the next biggest app, a few lucky developers hit it big, like Todd Moore, developer of **White Noise** and founder of **TMSOFT**.

Since learning to code at the age of 10, Moore has always been passionate about computer programming. When the original iPhone came out in 2007, he began creating his own apps which required jailbreaking the device. After official developer support was released by Apple, he began publishing his creations on the App Store. After launching the White Noise app, Moore experienced unexpected success. "It just instantly shot up in the ranks. I got calls from reporters, Jimmy Fallon made fun of it, Dr. Oz recommended it. It started generating money and I was able to quit my full-time job and start hiring people."

But his success has not been without challenges. During his early days as a creator, he found himself in a bit of legal trouble with a game in the iTunes store. "When I first started, I didn't register as a company. I published under my own name, Todd Moore." After releasing a game called Bubble Pop, a week later he found himself on the receiving end of a Cease and Desist letter. After settling that issue, Moore set up his own official LLC, and continues to warn other future developers of the hazards of self-publishing. "First tip, don't put it under your own name. There are a lot of legal issues in this field, and that is always my first bit of advice." He discovered that the more attention his products received, the more legal threats and patent trolls came knocking on his door.



While he is well-versed in many languages, his favorite language is C++. White Noise was mostly written in C++, which allows the code to be shared across many platforms including iOS, Android, Mac, and Windows.. He still codes as a founder, drawing upon his experience as a developer. "I kinda miss the days, back in 2008, with one version, one device. It was so much simpler then! (But) if I wasn't coding, I wouldn't be happy."

While building his company hasn't changed his love for coding, it has changed how he works in a team. "First thing I look for in people is passion. It's the number one thing. If I can detect that you're only in it for the money, it's not a good hire. I often ask if they're doing open source projects, anything on the side. Someone who enjoys it so much that they're doing it as a hobby and not just a career." Additionally, Moore has found interns to be a valuable asset when he started up his company. While initially unpaid internships, this mutually beneficial relationship allowed Moore to share his knowledge with the future of the industry and grow his company in the early days. "I pay them now," he jokes.

Throughout all stages and phases of his founder experience, Moore's best advice is to just get something out there. "It's more than software development, and there are so many things you have to learn along the way. Pick something simple, something easy that solves a problem, and just get it out there." As many developers turned founders have found, the hardest part of any journey is the first step. ■



The Code that Rocks the Cradle

by Kerri Couillard

Founder / Babierge

Life has a funny way of sending you down the path you are destined to travel. For me, this funny way involved two babies, a husband tired of living in the clutter of leftover baby gear, my professional capabilities as a developer and a growing entrepreneurial mindset.

I have an undergraduate degree in Fine Arts and an MS in Computer Science and Media Arts. I began my career as a Webmaster and Database Administrator. It was trial by fire and taught me to be a lifelong learner as well as the power of providing excellent, friendly service. Later, I worked as a Java Programmer and Web Application Developer.

When I was pregnant with my first son, I realized the flexibility of freelance work would be a better fit for my family and began working from home doing freelance Web Programming and Design. With the birth of my second son, I began to feel the call of entrepreneurship.



The Code that Rocks the Cradle, Cont.

As my sons grew, I ended up with a huge stack of leftover baby gear taking up a car bay in our garage. My husband, ready to park both cars in the garage again, suggested I sell it, but I thought I could rent it instead. I put up a one-page website and within 3 days I had my first \$300 reservation.

From there, I slowly hand built the Babierge Platform, embracing the newly-emerging Sharing Economy and watching delightedly as more and more reservations came in. It wasn't long before I started having bigger dreams. I wanted to build a brand focused on excellent customer service and grow Babierge to be an international company.

It can be scary to leave the security of a traditional job and head out alone into the startup world. If you're feeling this fear, you're not alone. Many Founders have been there. Trust yourself, believe in your work, and take pride in the core competencies you bring to your startup. Keep your eye on your customer. Solve a big problem. Change the way the world does something.

One of the toughest—and smartest—moves I made as an entrepreneur was replacing myself as CEO. Building a company is like raising children. There's a lot of letting go involved. I brought in Fran Maier (former Co-Founder of Match.com and TRUSTe) as CEO for Babierge, and together we are accelerating growth, building a brand, raising our first round of capital and sometimes it feels like leading a revolution.

Babierge is renting baby gear out of fifteen markets today, and we're adding new markets each month. We work with Independent Trusted Partners in local markets, helping them launch their baby gear rental businesses and providing entrepreneurial mentorship. We've rented close to 7,000 pieces of gear to date. We expect to add international markets in 2017.

What turns someone from a Developer to a Founder?

1. You get tired developing other people's dreams.
2. You discover a solution to a common problem.
3. The planets align (or your husband wants to park the car in the garage) and you jump in and give it your all, and then a little more, because that's what it takes—everything you have and then some.

The stakes are higher as an entrepreneur, but so are the rewards. I hope one day my children see that the dedication I've given to building a company that helps others is powerful. As the Founder of Babierge, it's my mission to help millions of families worldwide make lifelong memories as they vacation together.

I consider myself a developer at heart but now see myself as so much more. I'm a Founder. ■



The Twist at the Crossroads

Alexis Pratsides

Founder / MintTwist

The path from developer to founder is not always through a single app or product. Many developers take the path of putting their code to use for others. Alexis Pratsides, founder of [MintTwist](#), began his coding journey at a direct marketing company and later founded his own full service agency in 2002.

Many developers we talked to noted a key moment of decision. For Alexis, that moment came in the form of a layoff. "I trained as a software engineer and my first job was at an agency. I was working on large projects for blue chip companies. We were writing software for Dell to manage desktop applications. The business I was working for went through rapid growth and scaled their business up, but the clients didn't materialize, so I was laid off. I had to decide, do I get another job or do something for myself? I decided to give it a go."

No matter the industry, when you make the transition from developer to founder, there are things that must be left behind; sometimes, it's the very thing that brought you there. In a situation like Pratsides, you are no longer the developer in the business, but the founder. Pratsides notes, "Suddenly, you have the rest of the process to start thinking about and there's so much more you have to learn how to do." Accounting, pricing, marketing, it is all a side of business that he had never really seen.

"I miss just plugging in and coding all day. You love getting a kick out of solving a problem, fixing a bug, doing something really clever, the buzz of getting that project done; that's never going to go away. You do miss it." Although, he does note that you are rewarded in other, equal ways. Becoming a mentor, taking on community projects, or other tasks such as that can bring that same enjoyable experience to your new role.

His best advice when building a great team is to "hire people who have the same DNA as the business so the people who join are in sync. You can always upskill people, but if they don't fit in culturally, it's hard to make that work."

Ultimately, regardless of the journey and industry, Pratsides presents some sound advice if you're considering whether or not to take the leap. "So, if you're thinking about wanting to do it, there's a high chance you should give it a go at some point. Much better to have tried it, and if it doesn't work out, you can always go back." Whether you are hoping to found your own company based around a single app or software, or if you are wanting to build a full-service agency, the only thing you can do is give it a solid try. ■

When Your Backup Plan Becomes Your Biggest Success

Steve McConnell

Founder / Construx Software

In this growing age of technology, not everyone who becomes a developer starts out with that as their goal. Many start out with a different career or interest, only to find their way into the world of coding. Steve McConnell, founder of [Construx Software](#) and author, had this very type of change in his education that led him to where he is now.



McConnell originally pursued a degree in Philosophy, with Computer Science as a backup. When it came time to graduate, he took a job as a developer in the insurance industry. He found himself doing BASIC programming, APL, mostly GW BASIC. During this early point in his career, he had the opportunity to work on small, startup teams, and also large teams at Boeing and other smaller firms. He found it beneficial to see how teams of all sizes worked together, and it has shaped how he teaches teams in his consulting work now. "In all these environments, I saw things being done that were pretty dumb." In contrast, his work at Microsoft led him to better understand what it looks like when teams function very well together. "It was incredibly well run," he says of the giant company.

Many developers transition back and forth from developer to founder, however, McConnell's career transitioned between author and developer and then back again. In 1993, he published his first book, *Code Complete*, a 900-page book that took him two and a half years to finish.

While he enjoyed his time working at Microsoft, he realized he just couldn't work for anyone else. "I came to terms with the idea that the issue wasn't that companies were being run badly, but I didn't want to work for anyone else."

When McConnell started his own company, he did so by himself, but then he quickly began adding people, multiplying his force and eliminating the sole responsibility of it all. However, that came with its own issues. "The hardest part was the amount of time spent on staff issues. Staffing challenges and marginal performance takes a lot of energy out of me." Through the challenges of outsourcing in 2005 to the joy of watching his local team grow, it is all par for the course when you work with a team.

McConnell emphasizes knowing yourself as the key to building a successful business. "Know what your strengths and weaknesses are. If your skills are around development, you may not be the one to lead people. Hire people who are better than you at certain things in the business."

To this day, McConnell has been an advocate for understanding your own motivations. "Understand yourself really well. Why are you doing this? If you're doing it to get rich, that's a bad idea. That's an achievable objective for some, but at least get clear about what you want." From learning about sales and marketing, or just partnering with other diverse talent, you need to have many different tools in your belt in order to succeed. ■

3 Skills Devs Need NOW that They Didn't Need 5 Years Ago

The world of tech changes in a blink of an eye. When developing your own product and founding a company, your skills have to grow at the speed of that change. Successful developers-turned-founders gave us their top skills they believe are invaluable in this age of technology.

1. Skill Flexibility

With a rapidly growing tech market, developers have to stay on their toes and keep their skills sharp. Founder of SnapIT Solutions, Neelima Parasker, found flexibility to be key to long term success. "What's cool in the market may not be right - keep abreast of what's changing in the market." Keep on top of your game by attending webinars, subscribing to our newsletters, and never stop learning.

2. Code Bilingual

The more you know, the better off you will be. From C++, Python, and new developments such as Ruby, your language skills are going to be a prime asset. Todd Moore, developer and founder of the White Noise app, coded his most successful product in 4 different languages. "You have to learn more than one tool. The more languages, the more valuable you'll become. In the past you could specialize, but you're integrating a lot more now." Keep on top of your languages, and practice in some of the ones you don't use as often to keep your knowledge base fresh.

3. Marketing

A skill that was cited frequently by founders is marketing and sales. Developing a launching a product is only the first of many steps to success, and you have to appeal to the masses. Jeff Atwood, founder of Stack Overflow and Discourse, says that this is the top skill he would teach developers. "The art and science of building a product and the levels of purchasing. It's important to spend some time thinking about marketing." Unless you know how to sell, your business is only a hobby until people are buying.

Become a BuildBetter subscriber to get new and exclusive content for improving your code, team, and career.

[Subscribe Now](#)



The Second Time Around:

Repeating the Successes of the Past in a New Mission

Jeff Atwood

*Founder / Stack Overflow / Discourse.org
Co-founder / Stack Exchange*

When you finally reach your goal of a successful product launch, where do you go from there? Do you stick with it or sell it and move on and start the process all over again? Founder of Stack Overflow and co-founder of Stack Exchange, Jeff Atwood, experienced this dilemma after his product “accomplished its goal.” Between blogging and focusing on a new open-source project, Atwood’s experience is a path other developer-turned-founders may follow.

Repeating the Success of the Past in a New Mission, Cont.

In college, Jeff minored in Computer Science and graduated in 1992. He then moved to Colorado, took a few developer jobs “sort of pre-internet” (wha?), and later entered GlaxoSmithKline. While there, he had an epiphany—he knew he wanted to be somewhere where the software is the product. That led him to his next job at Vertigo Software. And while it was a great job and he was committed to the vision, he realized that he wanted to be part of the power structure of the company. And when a partner told him “I can’t go back in time and make you an owner of the company,” that was when Jeff knew he needed to become a founder.

“It’s what I wanted to do. So I reached out to Joel Spolsky because I wanted to do something with the blog.”

Up to this point, the only place for developers to talk about code was a website called Experts Exchange, a site that in his opinion “sucked.” His vision was a community where you could tell these little stories. Originally, it was “a way to get people to blog. When you tell the story, you understand it so much better.”

The business model of Stack Overflow was divided into two entities, Career and Q&A. Jeff, who was still actively developing the product, led the Q&A side of the business, while his partner ran the “job board” side. “It started making money and we started hiring people. Then a VC came along and then the company really started growing.”

Close to four years in, he’d been “getting cranky and burnt out at Stack.” In his family life, he had three kids, and life was changing. It appeared that they had solved their original problem that Stack Overflow set out to solve, because “no one remembers the old thing anymore [“Experts Exchange”]. We succeeded. The first part of the mission was complete.”

Jeff decided this was the perfect time to leave. However, during his time off, he began getting a little stir crazy and realized, “I needed a thing to do. I had to be on a Mission.”

While at Stack Overflow, he realized there was an unmet need. People would try to have a conversation or post something funny, and the system just wasn’t meant for that. Stack Overflow and



There’s a point where you know you’ve accomplished the goal. That may be the time to start something new. Like Jeff, take what you’ve learned about your first company, the successes, the failures and unmet needs, and build on top of that to create success in your next venture.

Stack Exchange were built for asking and answering questions. So Jeff created discourse.org, a system for having discussions. “It teaches you how to interact with people in a sustainable way and not hurt each other. The only action is a heart—it’s a system of empathy.”

One major difference in his new company is the fact that Jeff doesn’t code anymore. He wanted a software stack that was amenable to open source, so he made the decision that Discourse would be built in Ruby and JavaScript. That means that with Discourse, Jeff is doing more of the business-side tasks and getting more hands on with the sales. He admitted that learning to take money from people had been one of his biggest challenges, as his partner handled much of that in his previous company.

“Engineers are bad at taking money from people. Programmers aren’t that comfortable with the people. There is a gap to cross to try to be good at selling stuff.” He’s learned that “overcoming your own internal hurdles” is often the hardest part.

To overcome this in his new company, he has taken a different approach to sales. “My enthusiasm comes through, so it’s not about the selling.”

His advice to other developers looking to build a product company is to “get started on step zero. Try to launch the minimum viable thing. That experience of sucking at the first one makes you not suck at the next one. The sooner you start that feedback loop, the better.” ■

Developing for Change

The path from developer to founder takes people on many different paths. In the process of creating a non-profit volunteer pipeline and event management platform, Be a Doer founder John Negron discovered his experience to be a lot harder than expected.

Negron grew up in an immigrant family with not much initial interest in computers. While working at NASA and starting out in C++, Perl, and the .Net space, Negron was an avid volunteer in his community. During this time, he found that while organizations are great at serving people or communities in need, they struggled to engage volunteers in a meaningful and valuable way. Later, this experience would be the foundation for [Be a Doer](#).



Throughout his career as a developer, Negron has dabbled in start-ups and product launches before his own journey began.

"As a dev, the markets I've been in, there are a lot of people with great ideas. Every now and then, I get a pitch to come be the CTO of 'this'. I wanted to help, and I did here and there." However, after taking a hard look at his own passions and skillsets, he decided to focus on something that would feel less like work. After volunteering at a soup kitchen in NYC, the idea for Be a Doer really came to life. "It really just became a hobby, like, wouldn't this be cool." While working full time as a developer, he slowly began to transition into the founder role. "As a business person, it's not just about the product, or about a hobby. Getting into an accelerator program really helped me realize this." After a successful soft launch and seeing traction with the product, he decided to go all in.

John Negron

Founder / Be a Doer

In this process, Negron noticed the difference between technical founders and non-technical founders within the startup space. In some senses, he felt a distinct advantage in being a technical founder in the software development world. However, he found the most important thing is to partner with different skillsets to help compliment your own. "Getting sales and marketing, not my strong suit, that's the biggest thing. I'll be spending the next 5-10 years in this company, so choices of who I work with are as important as getting married."

Overall, there has been a prominent take-away from this experience. While every dev to founder journey is unique, one unifying factor stands out. Make sure that there is a market for your product or service. "The tendency is to build this thing and then they will come. You have to go against your own nature as a developer. Do something lightweight, maybe not even a product, but rather a landing page, or buy some ads and tell people about it and see if there's interest." There is a clear moment when your hobby becomes a viable product or service, and every future founder needs to be on the lookout for it. In Negron's own words, "It's not a business until someone is ready to pay for it." ■

Coded to Teach

Stephen Foster

Founder / ThoughtSTEM

You will be hard-pressed to find someone who doesn't enjoy or engage with video games in this modern age. From mobile apps, Facebook games, to everyday gamification, we are always at play. Developer turned founder, Stephen Foster, found his success in the gaming world by creating a way to help others learn to code. **ThoughtSTEM** was created to help people, particularly those under the age of 18, learn to write and develop their own code.

A challenge many founders experience is finding a unique sell, and Foster does just that with ThoughtSTEM. "We want to teach as many people as possible how to code, and the secret sauce is video games." With over a million programs written through their software, Foster used his passion to build a successful business. While there were ups and downs in his coding lifetime, his passion of teaching and learning computer science drove the creation of video game-based learning. "We just kind of lucked into it because we were bored in our lectures." After deciding not to go to college and falling into a get-rich-quick scheme, the positive takeaway was a drive to become his own boss. Although he did return to complete his education, and Foster now holds a Ph.D.

While many developers-turned-founders cite that it can be a struggle to have a technical background in a founder role, Foster sees it as an asset. "I've seen other founders who didn't have a technical background. They struggled to manage and communicate with their developers." The ability to speak the language to your developers is essential for a tech company. While other founders have mentioned marketing and sales as areas they may struggle with, technical communication is still key.

No matter what stage you are in your career, Foster gives this advice. "You should never lose sight of the fact that learning to code is a lifelong journey. There is always more to learn about computer science." As you grow closer to becoming a full-fledged founder, you may get pulled in many different directions, but don't forget to tend to your craft. Being a lifelong learner can only help you while you and your business grow.

<p22>



Call for Contributors

Do you have a story, resource, or insight that can help developers build better code, teams, and careers?

BuildBetter is both an eMag and a site hub. We're always open to content from you, like:

- Topic suggestions/general ideas
- Blog articles
- Interview subjects (volunteers, nominations)

[Help us BuildBetter](#)

</p22>

Rapid Dev Improvement

Developers & the Full Application Lifecycle

The most common reaction we get to our BuildBetter eMag is, "What is this?" True, we aren't a publishing company, but we are a developer company, and that's bigger than building software.

Applications are taking over the world, and the role of the developer is moving from the back room to the forefront of any business. To meet rapidly increasing demands, developers need to focus their time on what they do best: building amazing solutions. These are not only cool ideas, but tools that solve real problems as designed.

Stackify exists to increase developers' ability to create amazing applications as quickly as possible. We believe that the ideal development team, today and in the future, is consistently optimizing its output across the entire lifecycle of an application; from development to testing to production. Stackify's mission is to give

developer teams easy access to powerful tools, which enable them to take the lead in delivering the best applications as quickly as possible.

For more information on our developer tools, visit stackify.com.

For more dev-centric information and resources, visit our content hub, [BuildBetter Online](#).

And as always, we love to hear your comments and requests: mhobbs@stackify.com

Build on,



Max Hobbs

Stackify CMO

Get a free Stackify DevBox for your team/user group!

- 3 decks of our coveted Developers Against Humanity card game
- 3 of our limited edition developer t-shirts
- Free downloads of Prefix—a magical code profiler for .NET & Java
- Free licenses of Prefix premium—just for your members



[Request the Stackify DevBox](#)



BuildB**etter**
by Stackify