

# Issue 090: Startups

De Programmatica Ipsum

2026-03-02



# Contents

Issue 090: Startups	1
Flux	3
Exercise Those Stock Options, If You Can	7
futureform	13
Eric Ries	17
Jessica Livingston	21



## Issue 090: Startups



By Adrian Kosmaczewski, March 2nd, 2026

Welcome to the 90th issue of *De Programmatica Ipsum*, about *Startups*.

In this edition:

- Graham analyzes how startups could evolve<sup>1</sup> in our world of AI and agents.

---

<sup>1</sup><https://deprogrammaticaipsum.com/flux/>

- Adrian ponders whether working as a software engineer in a fast-paced startup<sup>2</sup> is worth the effort.
- In our Vidéothèque section<sup>3</sup>, we watch “The AI Startup Grift is Getting Worse” on the futureform<sup>4</sup> channel.
- In the Library section<sup>5</sup>, we review “The Lean Startup” by Eric Ries<sup>6</sup>, and “Founders at Work” by Jessica Livingston<sup>7</sup>.

Download this issue in DRM-free PDF<sup>8</sup> or EPUB<sup>9</sup> format, and read it on your preferred device. You can also subscribe to our RSS feed<sup>10</sup>, featuring the full content of our articles.

We would like to thank our patrons who generously contribute every month (or have contributed in the past) to our work and help us run this magazine. Thank you so much! In alphabetical order: Adam Guest, Adrian Tineo Cabello, Benjamin Sheldon, Christopher Nascone, Colin Powell, Franz Lucien Moersdorf, Guillermo Ramos Álvarez, Jean-Paul de Vooght, Dr. Juande Santander-Vela, Patryk Matuszewski, Paul Hudson, Quico Moya, Roger Turner, Szymon Licau, and countless more leaving anonymous tips every month.

Enjoy this issue! Please share our articles on social media, or contribute<sup>11</sup> if you would like to support our work with a donation via Liberapay<sup>12</sup>.

Cover photo by Mika Baumeister<sup>13</sup> on Unsplash<sup>14</sup>.

---

<sup>2</sup><https://deprogrammaticaipsum.com/exercise-those-stock-options-if-you-can/>

<sup>3</sup><https://deprogrammaticaipsum.com/category/videotheque/>

<sup>4</sup><https://deprogrammaticaipsum.com/futureform/>

<sup>5</sup><https://deprogrammaticaipsum.com/category/library/>

<sup>6</sup><https://deprogrammaticaipsum.com/eric-ries/>

<sup>7</sup><https://deprogrammaticaipsum.com/jessica-livingston/>

<sup>8</sup><https://deprogrammaticaipsum.com/pdf/issue-090-startups.pdf>

<sup>9</sup><https://deprogrammaticaipsum.com/epub/issue-090-startups.epub>

<sup>10</sup><https://deprogrammaticaipsum.com/index.xml>

<sup>11</sup><https://deprogrammaticaipsum.com/contribute/>

<sup>12</sup><https://liberapay.com/akosma/donate>

<sup>13</sup>[https://unsplash.com/@kommunikation?utm\\_source=unsplash&utm\\_medium=referral&utm\\_content=creditCopyText](https://unsplash.com/@kommunikation?utm_source=unsplash&utm_medium=referral&utm_content=creditCopyText)

<sup>14</sup>[https://unsplash.com/photos/white-and-black-wooden-quote-board-Y\\_LgXwQEx2c?utm\\_source=unsplash&utm\\_medium=referral&utm\\_content=creditCopyText](https://unsplash.com/photos/white-and-black-wooden-quote-board-Y_LgXwQEx2c?utm_source=unsplash&utm_medium=referral&utm_content=creditCopyText)

# Flux



By Graham Lee, March 2nd, 2026

We are at a point in history where, for the first time, it is possible for the tech startup as we know it to become a thing of the past. Of course, other kinds of startups are already things of the past, so it is not the *most* momentous of historical events. Software unicorn mega-deals are out, Forbes tells us<sup>15</sup>, and VC-funded AI startups are in. Medium-sized deals are more likely to be funded by risk-averse private equity than by

---

<sup>15</sup><https://www.forbes.com/sites/dariashunina/2025/12/16/the-vc-landscape-in-2025-key-trends-that-shaped-the-year/>

venture funds, and small seed deals have all but disappeared.

Even in the end times of the dot-com bust, the idea that internet companies would go away was inconceivable—yes, the funding went away for the most part, but businesses that had a clear benefit beyond “make a website” continued to get founded, and funded. In fact, web 2.0 hit the mainstream *after* the bubble had burst—social media, wikis, tagging sites, Google, WordPress. All of these companies sprang into existence as web-based startups in the post-dot-com slump.

Software teams—the people who make the things that make the money—reacted to the funding withdrawal by contracting their ambition and focusing their work on demonstrating their value. Working for years to create yet another groupware product was out, and working for weeks on a Minimum Viable Product was in.

This post-bubble “lean” stretch was also the era in which various lightweight development methodologies became badges of honour for virtuous software teams. Being “agile” showed that you were ready to adopt “eXtreme” practices (like talking to your customers, and prioritising giving them something they wanted). As a result, software development became more efficient and more focused. The next recession, in 2008, motivated more cost savings: renting servers instead of buying them, for example. Software startups changed a lot, but they did not go away.

But what happens this time? On the one hand, it is easier than ever to make a startup *product*: tell a model what the product should do, tell it again a couple of times<sup>16</sup> until it gets it right, then give the model your platform’s API key and tell it to do a Kubernetes.

That greater access to creating the *product* makes it much riskier to make a *business* around that product, though, as everybody else can do exactly the same. Someone who sees your product and thinks it should work slightly differently, or be slightly cheaper, can make that happen in little time. Particularly, any incumbent company who was far-sighted enough to bench underutilised staff instead of laying them off can quickly repurpose a whole team towards cloning your new thing, but with their logo and marketing budget and intellectual property counsel.

So does that mean no more software products? No, but it might mean way fewer software product *companies*. Really, they have had a good run. The era of startups that made computers for other people to make software for was relatively long-lived, from

---

<sup>16</sup><https://library.chironcodex.com/books/patterns-in-ai-augmented-software-development/page/record-prompt>

the Eckert-Mauchly Computer Corporation<sup>17</sup> in 1946 to, arguably, NeXT, Inc.<sup>18</sup>; the last of the 1980s workstation startups.

Then we had the era of startups that make use of computers to do things people want to do with computers; from the office/groupware companies that apply those workstations people were buying in the 1970s and 1980s (VisiCorp in 1977, and Lotus and Adobe in 1982 as examples), through to the apps and web apps that until recently could still attract decent levels of funding (Dropbox in 2007, ByteDance in 2012, Canva in 2013, and Figma in 2016).

Now, computers are part of the fabric of (a lot of) society, so “thing you can already do, but using a computer” is less of a draw. Generative AI and other neural network applications are helping to color in up to the edges on that picture, because they help open up the space from “things we can instruct a computer to do” to “things we cannot describe but can show the outcome of” and “things we can only explain by analogy to another thing”.

There are some avenues for a last hurrah for computers in themselves to remain interesting. Quantum computing allows for novel algorithms; blockchain for novel (computer-mediated) social interactions; virtual and augmented reality, and neural interfaces, for novel interaction paradigms; and increased “thingfulness” of the physical world for novel contexts. For example, I still wrote this draft the old-fashioned way, with a fountain pen on real paper, because that affords me a better experience than either typing or the current crop of stylus-on-glass and ballpoint-pen-that-digitizes-strokes tools.

However, those avenues should properly be seen not as opportunities to recover the “golden days” of software-for-software’s-sake, any more than pulse-code modulation revived Marconi-era radio mania or high-speed trains returned us to the glory days of Stephenson and Brunel. Computing is moving from “becoming” to “is”, and the next crop of new products will not ask “how do we exploit the technology”, but “what do people need and what do we use to give them that”. Computing will be a frequently-used component in the answer, but is no longer the answer for its own sake.

We are in the era of “things a computer can do for you that previously you had to do, whether or not you used a computer”. Personal accounting packages helped *you* to file your taxes; a personal accounting *agent* files your taxes for you. Whether it does it with a fad technology—object-oriented, web 2.0, genAI, blockchain—is immaterial, because

---

<sup>17</sup>[https://en.wikipedia.org/wiki/Eckert-Mauchly\\_Computer\\_Corporation](https://en.wikipedia.org/wiki/Eckert-Mauchly_Computer_Corporation)

<sup>18</sup><https://en.wikipedia.org/wiki/NeXT>

it does it autonomously. Computers have become boring, and that is the most exciting thing that has happened to computing in my lifetime.

Cover photo by Fiona Jackson<sup>19</sup> on Unsplash<sup>20</sup>.

---

<sup>19</sup>[https://unsplash.com/@fionakiwi?utm\\_source=unsplash&utm\\_medium=referral&utm\\_content=creditCopyText](https://unsplash.com/@fionakiwi?utm_source=unsplash&utm_medium=referral&utm_content=creditCopyText)

<sup>20</sup>[https://unsplash.com/photos/a-person-with-a-backpack-on-a-path-in-the-woods-cEmnjnv3ehY?utm\\_source=unsplash&utm\\_medium=referral&utm\\_content=creditCopyText](https://unsplash.com/photos/a-person-with-a-backpack-on-a-path-in-the-woods-cEmnjnv3ehY?utm_source=unsplash&utm_medium=referral&utm_content=creditCopyText)

# Exercise Those Stock Options, If You Can



By Adrian Kosmaczewski, March 2nd, 2026

The word “startup” became popular during the 1990s web craze. I do not remember it being a thing in Europe or Latin America before that time; of course that does not mean that startups did not exist, just that they were not in my radar as possible employment options for my adult life. They were certainly not in my parent’s radar, but somehow the press frenzy around the future being made in a proverbial garage called

their attention.

So that is how I started my career; joining a small web startup in the outskirts of Buenos Aires, one sunny day of October 1997. Said company aimed, following the words of its founder, to create an online marketplace for the ocean fisheries market.

And so I ended up joining their ranks, first from home<sup>21</sup>, and then in their offices. Let me for a moment stress the fact that I was literally working from home in 1998, with a dial-up internet connection and ICQ chat rooms, using a dedicated phone line paid for entirely by my employer. (Just think about this for a minute, and then reconsider the recent decision by your Founder & CEO who wants to build a “family” in a big noisy open space, and thereby dictates mandatory time in the office for all employees. Do not worry, I will wait.)

The first thing that comes to my mind when thinking about startups is, of course, a certain taste for transgression and breaking the rules. Working for a startup meant bending the world, for good or for worse, and trying new things bypassing a shallow hierarchy, in a way that would be unthinkable in a more standard corporate setting.

This trend accelerated during the Web 2.0 era, and later during the social media, mobile, blockchain, and (of course!) AI gold rushes. Every one of these waves brought its own flavor of startups and unheard benefits and/or colorful workspaces, filled with foosball and ping-pong tables, slides, post-it notes on every whiteboard, and (more recently) matcha latte baristas.

Already in 1997 I could feel the “agility”, the “cash burn rate”, and the “time to market” pressures on my young shoulders; mind you, none of these buzzwords had yet been invented (again, not that I remember anyway). The use of a non-compiled, typeless ~~and shitty~~ programming language like VBScript fit the bill perfectly well in those contexts. We delivered as much code as possible, testing be damned (to be frank, we had not yet heard about Kent Beck<sup>22</sup> at the time, and we were not even using an SCM, not even CVS<sup>23</sup>).

At a time when everybody else was building Windows apps in Visual Basic 6, we were building an intranet for our employer in VBScript. Can you feel it? This is exactly the sensation of so many startups at the heyday of each technology wave of the past 30 years:

---

<sup>21</sup><https://akos.ma/blog/wfh-in-1998/>

<sup>22</sup><https://deprogrammaticaipsum.com/kent-beck/>

<sup>23</sup><https://deprogrammaticaipsum.com/twenty-years-is-nothing/>

- When everybody was using boring thick Windows laptops, that startup was writing their software on a shiny, thin-as-paper MacBook Air.
- While everybody was setting up servers and installing operating systems, another startup entered their credit card on the website of a cloud provider and started provisioning servers 20 minutes later, thereby replacing CAPEX<sup>24</sup> with OPEX<sup>25</sup> and thrilling their investors.
- Then, when most developers were restarting their Tomcat application servers, another group of idealists typed the command `rails new` and bet their future on an unknown scripting language from Japan<sup>26</sup>.
- At a time when everybody had a BlackBerry in their pocket, some other startup was building a new “iPhone OS” application (kids: that is how iOS was named from 2007 to 2010) and hoping for Facebook to buy them.
- Later when everybody was still writing code manually, a group of friends in a garage started using Cursor and Claude Code to rewrite the world from scratch.

Ah, the sweet feeling of transgression; the essential component of any respectable startup.

I met people who worked in certain startups where a weird thing called “stock options” was part of their compensation plan. This is how they are supposed to work: instead of paying you with money, they give you a paper that states that IF the company did not go bust and IF they decided to go public in some market (most probably the NASDAQ) and IF the tight time deadlines (and other shenanigans written in small print) for such operation are respected, THEN the owner of the document was entitled (but not obliged, this is an “option” after all) to buy some stock at a fixed price, and then by selling it, finance a home, a vacation in Papeete, or, yes why not, another startup.

I forgot to mention another interesting IF in the equation above: one should “execute the option” only IF the price of the stock was higher than the strike price written on the paper; otherwise, you would be buying a worthless asset, which is never a good idea.

Count the amount of IFs in the previous paragraphs, and you will realize why you do not know many people that got rich using this scheme (outside Silicon Valley, that is). In essence, the idea of stock options on a startup seems enticing to two kinds of people: to employers, because they can discount on actual salary costs, and for junior software engineers, who did not take a minor in finance and have no clue how the stock market

---

<sup>24</sup>[https://en.wikipedia.org/wiki/Capital\\_expenditure](https://en.wikipedia.org/wiki/Capital_expenditure)

<sup>25</sup>[https://en.wikipedia.org/wiki/Operating\\_expense](https://en.wikipedia.org/wiki/Operating_expense)

<sup>26</sup><https://deprogrammaticaipsum.com/issue/issue-089-ruby/>

or derivatives work. A win-win situation for everyone.

But dreams are cheap, and everyone wants to become rich overnight.

The truth is that most startups go bust: according to Forbes<sup>27</sup>, purveyor of wisdom in the science of making money, 20% of startups die within the first year, and 65% within 10 years. For junior software engineers, that means that there is a 65% chance to be jobless at the age of 35. Actually it is a bit worse than that: you will be holding a name in your CV of a company long gone, which did nothing relevant or memorable, until the day you retire (I know a thing or two about that, as I have at least 4 names in my own résumé that follow this pattern).

Why do those startups fail? There are countless Harvard Business Review<sup>28</sup> articles about that, but I will not link you to any of them; instead, I will provide you with my highly questionable theory about the root issue at the heart of most startups: *the unbearable hubris levels of their founders*. Through the decades I have had the unfortunate experience of crossing paths with quite a few psychopaths who, in between meetings with potential investors and micromanaging their engineers, had the inner belief and desire of being the next Steve Jobs.

Reality proved some of them otherwise. Yes, not all; some were lucky enough to actually make it through their “exit” and “cash out” strategies unscathed, which can not always be said of their employees. But that is another story.

Because a sad side effect of the infatuation with startups was the slow degradation of any work security. Gone are the days when a software engineer would spend 35 years at IBM; welcome to the new age of job hopping and burnout. Too bad if your health takes a toll and you are dismissed tomorrow for whatever whimsical reason. Apparently this new state of affairs is a good thing, because the sacrosanct “free market” guarantees higher wages and new experiences. Yay!

The truth is bleaker than that. Bashing worker unions<sup>29</sup>, brainwashing junior engineers with the idea that “we are a family” and other atrocities, screaming against regulation and oversight in the name of free markets; all of these ideas have caused our social tissue to degrade and decompose. The latest waves of massive layoffs in IT, most probably poised to continue thanks to the inexorable advances in AI, are the living proof that workers, more than ever, are fodder ready to be consumed and promptly discarded, for

---

<sup>27</sup><https://www.forbes.com/advisor/business/software/startups-failure-rate/>

<sup>28</sup><https://hbr.org/>

<sup>29</sup><https://deprogrammaticaipsu.com/issue/issue-042-trade-unions/>

the profit and benefit of yet another Series D round that would make your startup a shiny unicorn on the top page of [startupticker.ch](http://startupticker.ch)<sup>30</sup>.

Startups are here to stay. For a software developer, I do believe that working at a startup is still an excellent way to learn, experiment, and take responsibilities early. And who knows, maybe *your* stock options might be worth something, someday. But as always, *caveat lector*; try to gauge the founder intentions as early as possible. You have been warned.

Cover photo by Maxim Hopman<sup>31</sup> on Unsplash<sup>32</sup>.

---

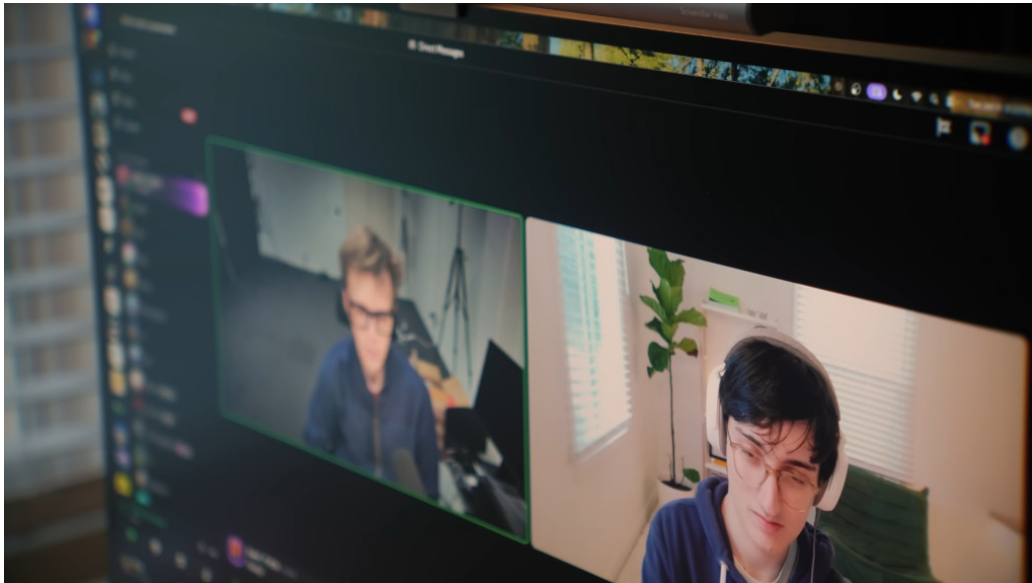
<sup>30</sup><https://www.startupticker.ch/en/news>

<sup>31</sup>[https://unsplash.com/@nampoh?utm\\_source=unsplash&utm\\_medium=referral&utm\\_content=creditCopyText](https://unsplash.com/@nampoh?utm_source=unsplash&utm_medium=referral&utm_content=creditCopyText)

<sup>32</sup>[https://unsplash.com/photos/red-and-blue-light-streaks-fiXLQXAhCfk?utm\\_source=unsplash&utm\\_medium=referral&utm\\_content=creditCopyText](https://unsplash.com/photos/red-and-blue-light-streaks-fiXLQXAhCfk?utm_source=unsplash&utm_medium=referral&utm_content=creditCopyText)



## futureform



By Adrian Kosmaczewski, March 2nd, 2026

To say that we live in a pivotal moment in tech history is such commonplace that you would think it would be unworthy to use such an epithet in this journal. Yet we do think that, but we are very aware that it will be only a decade or so before we can perform a conscious analysis of the various messes of our era (if this magazine is still around, that is). In the meantime, we have hunch feelings and commentators.

Witnessing history unfold is a temptation for speculation and cheap commentary; there is, however, some interesting points to make, some of which constitute the essence of this month's Vidéothèque entry: "The AI Startup Grift is Getting Worse"<sup>33</sup> in the

---

<sup>33</sup><https://www.youtube.com/watch?v=dyGZNU0XsYg>

recently-started YouTube channel futureform<sup>34</sup>.

The core of the video is the analysis of (yet another) AI startup promising a future that could never be delivered (or at least not using 2026 technology). In this case, it is about a set of “augmented reality” glasses (clearly, a fad that has been refusing to die for more than 15 years at this point) called “Pickle”.

The startup behind this umpteenth marvel of technology claims that their AR glasses are smaller, more lightweight, with longer battery life, and with features that “disrupt” (gotta love those words in the startup world, innit) whatever Meta and Apple have been able to put together, their oversized budgets and hordes of PhDs be damned. One cannot but wonder at the level of delusion at play.

Do I need to continue? Not really; you know where this is heading. It is more than vaporware, it is the sign of an overheated bubble of undefendable claims, covered by nothing else than the sound of an echo chamber that keeps getting bigger and bigger.

Sadly, the “AI Grift” culture is here to stay, just like startups. There is a lot of easy venture capital cash to be had just by voicing these ridiculous thoughts out loud, regardless of common sense and the actual existence of high-profile failures *in the same technology space*, like the Humane AI Pin<sup>35</sup>, the Rabbit R1<sup>36</sup>, and other recent embarrassments in the same market. If all else fails, at least listen to what Marques Brownlee<sup>37</sup> had to say about them.

We congratulate the host of the futureform channel<sup>38</sup> for performing a thorough research and analysis of the current state of the art of bullshit in the AI startup world, and how this is all getting out of hand one press release at a time. Other videos in the channel provide interesting insight about mobile devices, self-driving cars, and more. Clearly worth a click on the “subscribe” button, if you ask me.

Enrique Santos Discépolo wrote in 1934 one of the most famous tangos of all time, titled “Cambalache”<sup>39</sup> (a word that in Rioplatense Spanish<sup>40</sup> means “junkyard”), whose censored lyrics openly criticized the corruption and hubris of the era, with timeless words that kept sounding for almost a century with uncanny relevance.

<sup>34</sup><https://www.youtube.com/@future-form>

<sup>35</sup><https://www.youtube.com/watch?v=TitZV6k8zfA>

<sup>36</sup><https://www.youtube.com/watch?v=ddTV12hErTc>

<sup>37</sup>[https://en.wikipedia.org/wiki/Marques\\_Brownlee](https://en.wikipedia.org/wiki/Marques_Brownlee)

<sup>38</sup><https://www.youtube.com/@future-form>

<sup>39</sup><https://en.wikipedia.org/wiki/Cambalache>

<sup>40</sup>[https://en.wikipedia.org/wiki/Rioplatense\\_Spanish](https://en.wikipedia.org/wiki/Rioplatense_Spanish)

Hoy resulta que es lo mismo  
 ser derecho que traidor,  
 Ignorante, sabio, chorro,  
 generoso o estafador  
 Todo es igual, nada es mejor  
 Lo mismo un burro que un gran profesor

In English:

Today it turns out that it's the same  
 to be honest or a traitor,  
 Ignorant, wise, thief,  
 generous or swindler  
 Everything is the same, nothing is better  
 A donkey is the same as a great teacher.

The tech startup world, led by an ever-increasing horde of clueless “AI bros” is entering a critical phase, sadly in tune with the overall sense of hopelessness one has after learning the (admittedly abysmal) daily news of our world.

Take a deep breath, and watch this month’s Vidéotheque entry “The AI Startup Grift is Getting Worse” on YouTube<sup>41</sup>.

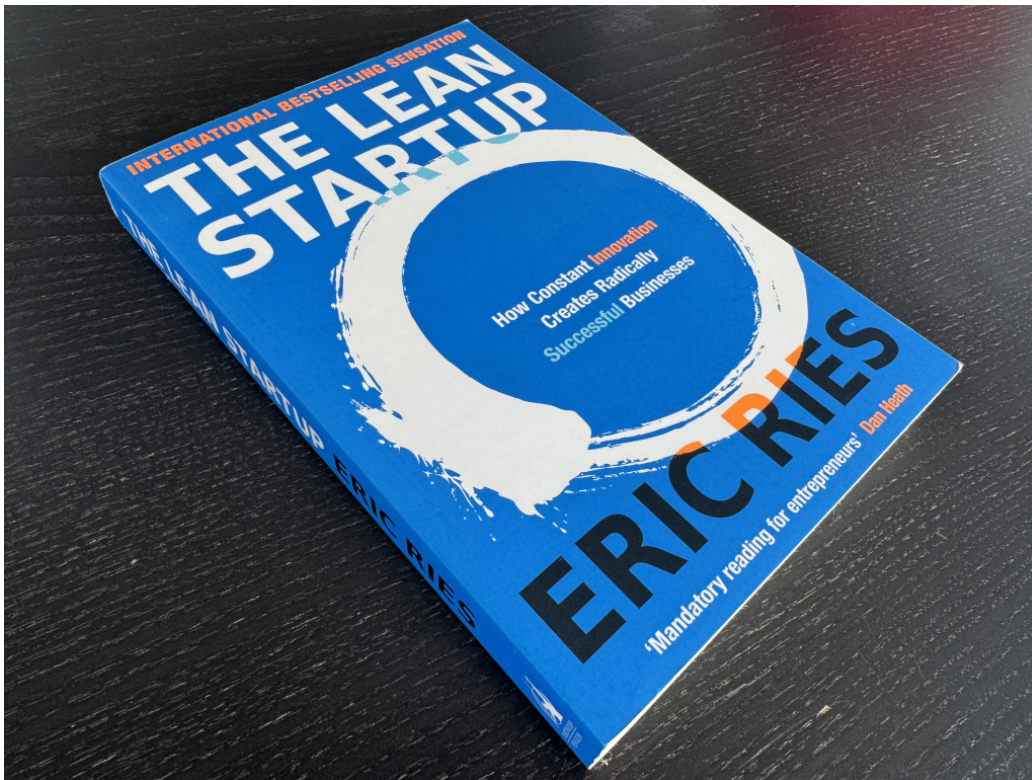
Cover snapshot chosen by the author.

---

<sup>41</sup><https://www.youtube.com/watch?v=dyGZNU0XsYg>



# Eric Ries



By Graham Lee, March 2nd, 2026

What if I told you that your startup can avoid making a product, it just needs to suggest that it might make a product and see if anybody agrees to buy it? You might think your humble author to be mad. What if I told you that a former CTO of a startup that never managed an exit wrote that on a blog, and then again in a book? OK, *now* the idea is plausible!

That is not, of course, the message behind “*The Lean Startup: How Today’s Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*”, but it is the part of the story that resonated the loudest. Eric Ries had spent a long time with his company, IMVU<sup>42</sup>, building a 3D avatar experience for chat clients that it turned out people did not want to use. His insight was that the company could have discovered much earlier that people did not want to use it. They did not need to build it to discover that; they just needed to ask.

Coming first in blog form in 2008, and then book form in 2011, “*The Lean Startup*” distilled lessons from other authors, notably Steve Blank<sup>43</sup>, into pithy, actionable advice. More serendipitously it landed at the perfect time for investors and technologists looking for advice on how to make technology companies on the cheap. The dot-com bubble’s burst was still in recent memory, with VC investment hitting its lowest ebb in 2003. Venture capital availability had turned down again in 2005, with investors preferring to support later-stage companies, and then 2007-2009 brought the Great Recession, with the property-led Global Financial Crisis at its peak in 2008. In short, there was not any money, so you may as well spend what little you had on a book that told you how to make do without any more.

The core message of the book is “pivot or persevere”: gather data that validates or invalidates your current course and *pay attention to it*. If customers are not excited by what you are promising, promise something else. If people react well to it, then carry on—but look for the next opportunity to test your assumptions.

In this, we see the seeds of some of the great technology companies of our time. For example, Justin.tv<sup>44</sup> started as a YouTube-like video broadcast site, but pivoted to Twitch.tv<sup>45</sup> in 2011 when they realised that videogame streaming was their most popular category. Burbn was a mobile check-in app created in late 2009, until the founders realised that their product was too similar to FourSquare, jettisoned all of the features except the popular photo-sharing, and chose a new name for the service they relaunched in 2010: Instagram.

The lean startup approach also brings many phrases that are in common startup (and other business) parlance today. A brief list:

- Continuous deployment: ship your experiments (and their outcomes) to pro-

---

<sup>42</sup><https://about.imvu.com/>

<sup>43</sup>[https://en.wikipedia.org/wiki/Steve\\_Blank](https://en.wikipedia.org/wiki/Steve_Blank)

<sup>44</sup><https://en.wikipedia.org/wiki/Justin.tv>

<sup>45</sup>[https://en.wikipedia.org/wiki/Twitch\\_\(service\)](https://en.wikipedia.org/wiki/Twitch_(service))

duction as quickly and as often as possible. Yes, we already had continuous delivery in the manifesto for agile software development<sup>46</sup>, but this time we *mean* continuous and we *mean* delivery; we do not just mean running your tests when you make a commit.

- Minimum viable product: identify the smallest thing you can build that gives you high-value information on whether your approach is a good one, and build that and get it to customers.
- Split testing: now more commonly called A/B testing, this is the practice of randomly assigning people into different groups and giving them different experiences, comparing the outcomes on some important metric and searching for statistically significant differences.

With hindsight, we see the development of the modern, cash-efficient startup. VCs can make small bets on early-stage companies, and reward those who learn and adapt the most to take into account the information they gain. Customers can get earlier information about what a company's planning to do, and decide whether that is valuable for them—along with earlier information about when the company decides to pivot away from doing that.

Of course, with hindsight, we also see the seeds of surveillance capitalism and enshittification. When your experiments show that it is easier to grow a free product with ads than a paid-for product. When A/B tests show that people interact more with revenue-generating ads if you design your product in this way.

We also see the era of small bets. If you do not get a result in a couple of weeks, pivot away. Do what customers react well to *today*, not what they might be absolutely clamouring for in a year's time, or a decade's.

It is not the only way to run a business, it is just the only way that gets funded.

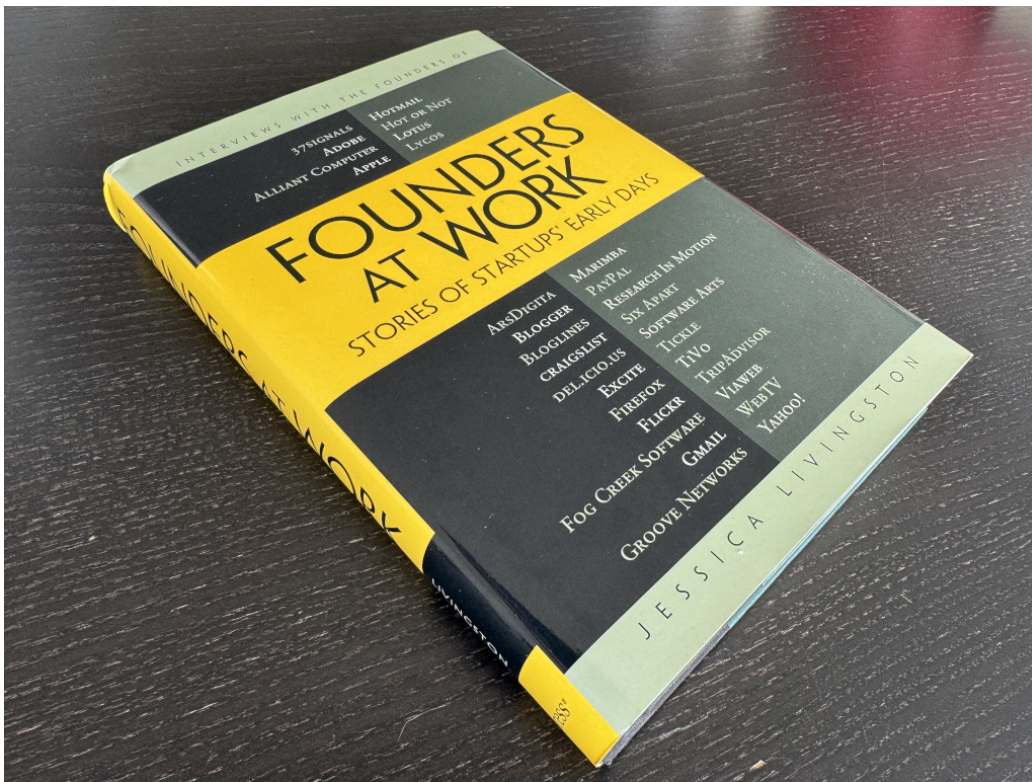
Cover photo by Adrian Kosmaczewski.

---

<sup>46</sup><https://agilemanifesto.org>



# Jessica Livingston



By Adrian Kosmaczewski, March 2nd, 2026

In many ways, the year 2007 was a crossroad in tech, and this has much more to do than just the introduction of the iPhone (although, by all means, that was quite a watershed moment). 2007 was the last moment in time our software workmanship operated without Git or GitHub. Without the Go programming language. Without Y Combinator.

Without Stack Overflow. Without Android. Without new JavaScript frameworks every weekend. Without Twitter. Without DevOps. Without Palantir. Without Instagram influencers. Without Docker containers or Kubernetes. Without Claude Code. Such was *l'air du temps* captured in the pages of one of our Library choices for this month: “Founders at Work: Stories of Startups’ Early Days”<sup>47</sup> by Jessica Livingston.

Well, to be honest Y Combinator<sup>48</sup> did exist already at the time of its publication, but it had not yet reached the level of relevance that it would end up having today. Just to give you an idea, Sam Altman, the very founder of OpenAI, quickstarted his own career with a “geosocial networking” website called Loopt<sup>49</sup> in 2005 (those with enough memory will surely remember) following a seed investment by Y Combinator. And 20 years later, here we are.

It is hard to speak about famous software startups today without mentioning Y Combinator. The list of startups funded in one way or another by it is too long to mention<sup>50</sup> here. Suffice to throw some names like Dropbox, Heroku, Justin.tv (later known as Twitch), Reddit, Quora, Stripe, Scribd, Bitnami, DoorDash, GitLab, Zapier... You get the idea.

The connection between Y Combinator and “Founders at Work” is simple: the author of the latter, Jessica Livingston<sup>51</sup>, is one of the co-founders of the former, and incidentally the wife of Paul Graham<sup>52</sup>, of Viaweb<sup>53</sup> and “On Lisp”<sup>54</sup> fame... whose interview also appears in the book on page 205.

For rather obvious timing reasons, this book does not dive into the anecdotes of Y Combinator’s own founders, but rather represents quite a large scan across various famous companies who all started in some proverbial garage with some crazy idea behind. The names of Apple, Research in Motion, Lotus, Adobe, and Software Arts (of VisiCalc fame) appear in these pages, through the words of their own founders.

The message behind the book was strong and clear: start your own company, no matter what. Y Combinator found a simple formula: provide enough founding for a startup

---

<sup>47</sup>[https://en.wikipedia.org/wiki/Founders\\_at\\_Work](https://en.wikipedia.org/wiki/Founders_at_Work)

<sup>48</sup>[https://en.wikipedia.org/wiki/Y\\_Combinator](https://en.wikipedia.org/wiki/Y_Combinator)

<sup>49</sup><https://en.wikipedia.org/wiki/Loopt>

<sup>50</sup>[https://en.wikipedia.org/wiki/List\\_of\\_Y\\_Combinator\\_startups](https://en.wikipedia.org/wiki/List_of_Y_Combinator_startups)

<sup>51</sup>[https://en.wikipedia.org/wiki/Jessica\\_Livingston](https://en.wikipedia.org/wiki/Jessica_Livingston)

<sup>52</sup>[https://en.wikipedia.org/wiki/Paul\\_Graham\\_\(programmer\)](https://en.wikipedia.org/wiki/Paul_Graham_(programmer))

<sup>53</sup><https://en.wikipedia.org/wiki/Viaweb>

<sup>54</sup>[https://en.wikipedia.org/wiki/On\\_Lisp](https://en.wikipedia.org/wiki/On_Lisp)

to become, following Paul Graham's own words, "ramen profitable"<sup>55</sup>, and grow from there. Of course this formula applies particularly well to a certain demographic: young people.

Ramen profitability is the other extreme: a startup that becomes profitable after 2 months, even though its revenues are only \$3000 a month, because the only employees are a couple 25 year old founders who can live on practically nothing.

Too bad if you are older than that, or if you have kids that, you know, need something else in their diet than ramen, which is usually the case. In those cases you had better stay at your corporate job, *thankyousomuch*.

Of course I am being cynical here. Mr. Graham describes, however, a couple of interesting aspects about "ramen profitability", in particular this one, often overlooked by most founders:

Raising money is terribly distracting. You're lucky if your productivity is a third of what it was before. And it can last for months.

This is also one of the reasons having a minimalistic team of at least 2 founders is also helpful, with at least one of them taking care of money, and the other(s) taking care of tech. Again, Paul Graham had something to say<sup>56</sup> about these mechanics:

Ideally you want between two and four founders. It would be hard to start with just one.

"Founder duo" pattern notwithstanding, the myth of the sole founder prevails in history books and popular culture; "Founders at Work" features interviews with just one founder for each hallmark startup in her book. (A nice touch, however, is that it is Woz<sup>57</sup>, and not Steve Jobs, who got to talk about Apple; although I can only imagine that Jobs was quite busy with the impending launch of the iPhone at the time of the preparation of this book, and promptly declined the request.)

It is time for a mandatory *caveat lector*: this book could have also been named "Mostly White Male Founders at Work". There are just three female founders in the pages therein, out of... 32 interviews. We are talking about Caterina Fake<sup>58</sup>, founder of

---

<sup>55</sup><https://paulgraham.com/ramenprofitable.html>

<sup>56</sup><https://www.paulgraham.com/start.html>

<sup>57</sup>[https://en.wikipedia.org/wiki/Steve\\_Wozniak](https://en.wikipedia.org/wiki/Steve_Wozniak)

<sup>58</sup>[https://en.wikipedia.org/wiki/Caterina\\_Fake](https://en.wikipedia.org/wiki/Caterina_Fake)

Flickr<sup>59</sup>; Ann Winblad<sup>60</sup> of Hummer Winblad Venture Partners; and Mena Trott<sup>61</sup>, co-founder of Six Apart. Also, all of them are either of American, Canadian, or European origin, except Sabeer Bhatia<sup>62</sup>, the founder of Hotmail.

Despite this lack of diversity, “Founders at Work” is a delightfully crafted collection of anecdotes from the trenches, war stories told by the generals that triumphed. It is, like any history book, biased: we only get the winners to tell their story. But many of those winners almost did not make it, and it in those cases that the book shines.

Suffice to mention the shenanigans of launching a blogging business during the dot-com bubble crash<sup>63</sup>, as told in page 120 by Evan Williams<sup>64</sup>, a man who would later launch Medium, and who was, precisely at the time of his interview, bootstrapping a new Ruby on Rails site called Twitter, which you might have heard about.

That was a really bad time. Actually the day that everyone told me they were leaving... I told everyone they were laid off and said, “Work with me if you can.” And at the time, everyone had already missed one paycheck, and they’d had it.

Ouch. So, just like the previous quote, the core idea of the book is anecdotes. Launch day issues. Growing pains. Marketing blunders. Lost financing. Coins dropping too late. There are so many learnings to get, it is hard to pinpoint the most representative. Let us try, anyway; spoiler alert: it ain’t pretty.

It’s hard to under-promise and over-deliver when everyone’s promising things for you. We’re trying not to hype up what we’re doing until we’ve got something people can use. People expect the world, so if you hype up what you are doing, you have to deliver, and it’s not easy.

(Blake Ross<sup>65</sup>, co-creator of the Firefox browser, on page 398.)

I could not sleep well for 4 years. If you are in charge of technology at a really fast-growing company that gets lots of publicity, there’s always something that worries you. In early 2000, it was scalability.

---

<sup>59</sup><https://en.wikipedia.org/wiki/Flickr>

<sup>60</sup>[https://en.wikipedia.org/wiki/Ann\\_Winblad](https://en.wikipedia.org/wiki/Ann_Winblad)

<sup>61</sup>[https://en.wikipedia.org/wiki/Mena\\_Grabowski\\_Trott](https://en.wikipedia.org/wiki/Mena_Grabowski_Trott)

<sup>62</sup>[https://en.wikipedia.org/wiki/Sabeer\\_Bhatia](https://en.wikipedia.org/wiki/Sabeer_Bhatia)

<sup>63</sup>[https://en.wikipedia.org/wiki/Dot-com\\_bubble](https://en.wikipedia.org/wiki/Dot-com_bubble)

<sup>64</sup>[https://en.wikipedia.org/wiki/Evan\\_Williams\\_\(Internet\\_entrepreneur\)](https://en.wikipedia.org/wiki/Evan_Williams_(Internet_entrepreneur))

<sup>65</sup>[https://en.wikipedia.org/wiki/Blake\\_Ross](https://en.wikipedia.org/wiki/Blake_Ross)

(Max Levchin<sup>66</sup>, co-founder of PayPal, on page 12)

You never know anything. The hardest part in a startup is that you wake up one morning, and you feel great about the day, and you think, “We’re kicking ass.” And then you wake up the next morning, and you think “We’re dead.” And literally nothing’s changed.

(Joe Kraus<sup>67</sup>, co-founder of Excite<sup>68</sup>, which admittedly depending on your age you might not have heard about until today, page 67.)

It all sounds very charming. Let us throw this assertion by Ms Livingston herself in the mix, found in the introduction:

Innovations seem inevitable in retrospect, but at the time it’s an uphill battle.

There are two parting thoughts I would like to add before closing this article.

The first one is that, if you are in the process of starting your own technology startup, please receive our most sincere wishes of success and strength, and our recommendation of reading “Founders at Work” in your (little) free time. But we would also have you keep in mind that being an entrepreneur is not for everyone, and pivoting, or simply closing your business, might be the right answer at the right moment depending on your own circumstances. Do not trade your mental health (nor that of your colleagues, co-founders, and employees) for the cover of Forbes. From the vantage point of the big scheme of things, it is simply not worth it.

The second thought is that, in the current times of Gödel’s Loophole being proven real<sup>69</sup>, Switzerland (and, in general, Western Europe) have a one-in-a-lifetime opportunity to snatch some talent from the other side of the Atlantic; and certainly, local organizations like startupticker.ch<sup>70</sup>, the EPFL<sup>71</sup>, or ETH Zurich<sup>72</sup> are doing all they can (sometimes, sadly, with little or no support from authorities) to make sure those minds find a location where to think in peace, launch, grow, and maybe IPO the next

---

<sup>66</sup>[https://en.wikipedia.org/wiki/Max\\_Levchin](https://en.wikipedia.org/wiki/Max_Levchin)

<sup>67</sup>[https://en.wikipedia.org/wiki/Joe\\_Kraus](https://en.wikipedia.org/wiki/Joe_Kraus)

<sup>68</sup>[https://en.wikipedia.org/wiki/Excite\\_\(web\\_portal\)](https://en.wikipedia.org/wiki/Excite_(web_portal))

<sup>69</sup><https://deprogrammaticaipsum.com/william-aspray/#:~:text=loophole>

<sup>70</sup><https://www.startupticker.ch/>

<sup>71</sup>[https://en.wikipedia.org/wiki/%C3%89cole\\_Polytechnique\\_F%C3%A9d%C3%A9rale\\_de\\_Lausanne](https://en.wikipedia.org/wiki/%C3%89cole_Polytechnique_F%C3%A9d%C3%A9rale_de_Lausanne)

<sup>72</sup>[https://en.wikipedia.org/wiki/ETH\\_Zurich](https://en.wikipedia.org/wiki/ETH_Zurich)

unicorn<sup>73</sup>. This book, and even more important, the founding model imagined by Y Combinator, should be inspirations for the Swiss community to increase the attractiveness of the local startup scene.

Cover photo by the author.

---

<sup>73</sup>[https://en.wikipedia.org/wiki/Unicorn\\_\(finance\)](https://en.wikipedia.org/wiki/Unicorn_(finance))