

Mission Built

Lessons from the Barbell and
the Boardroom



by Mike Nichols

Mission Built

Lessons from
the Barbell and
the Boardroom

Mike Nichols

Author of *Mission Built*

Veteran. Builder. Coach. User-Obsessed.

Mike Nichols has spent his life at the intersection of discipline and service—from military intelligence to software leadership to strength coaching. Across every domain, one truth has held: real progress starts when you care deeply about the person you're building for. The user isn't a persona or a number—they're the mission.

Mission Built is an open-source philosophy forged through reps in the gym and lessons from the field. It's for anyone who believes progress isn't pretty, leadership is earned, and showing up with purpose changes everything.

Follow along at missionbuilt.io

This work is licensed under a Creative Commons Attribution-NonCommercial 4.0 International License.

creativecommons.org/licenses/by-nc/4.0/

License

Mission Built: Lessons from the Barbell and the Boardroom
by Mike Nichols

Copyright © 2025 Mike Nichols

This work is licensed under the **Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License**.

To view a copy of this license, visit creativecommons.org/licenses/by-nc-sa/4.0

You are free to:

- **Share** — copy and redistribute the material in any medium or format
- **Adapt** — remix, transform, and build upon the material

Under the following terms:

- **Attribution** — You must give appropriate credit, provide a link to the license, and indicate if changes were made.
- **NonCommercial** — You may not use the material for commercial purposes.
- **ShareAlike** — If you remix, transform, or build upon the material, you must distribute your contributions under the same license.

No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

For permissions beyond the scope of this license, please contact:

mike@missionbuilt.io

The Loadout

License.....	4
The Loadout	5
Prologue - Built for More	8
Mission Before Metrics.....	9
The Mission Is the Magnet.....	9
The Drift.....	16
Repetition with Intention	19
Built Through Reps	24
The Myth of Overnight Success	24
Repetition Is Not Redundancy	27
When the Spark Fades.....	30
The Multiplier of Boring Work.....	33
The Work Becomes the Win	36
Rituals Over Rules.....	37
The Ritual Is the Rail.....	37
Listen to Your Signals.....	39
Change the Pattern, Not the Practice.....	41
Feedback Is A Superpower.....	45
Cues, Not Critiques	45
Make Feedback a Ritual	51
Listening Is a Lift.....	57
Strong Feedback, Strong Foundations.....	62
Progress Isn't Pretty	69
Mastery Is Repetition with Feedback	69
The Grind That Grows	73
Change the Phase, Not the Process.....	77
Growth That Holds	79
The Mission Demands Recovery.....	82
The Biology of Bounce Back	82

Boundaries Are Barriers to Burnout	88
Sustainable Strength.....	90
Aside: Recovery Creates Room for Perspective.....	93
Train the Engine, Not Just the Output	97
Work Capacity Over Window Dressing	97
Internal Strength for External Load	101
What Powers You?	107
Decisions Are Made Under Load	112
Stress Tests the System.....	112
Clarity Beats Certainty.....	116
Training for Chaos	121
Ship It Like You Show Up	128
Integrity in Small Things.....	128
The Shipping Standard.....	132
Everything Built In.....	137
You're Just Getting Warmed Up	143
Further Reading	144
Log 1 - Mission Before Metrics.....	144
Log 2 - Built Through Reps	147
Log 3 - Rituals Over Rules	150
Log 4 - Feedback Is A Superpower.....	152
Log 5 - Progress Isn't Pretty	155
Log 6 - The Mission Demands Recovery.....	157
Log 7 - Train the Engine, Not Just the Output.....	159
Log 8 - Decisions Are Made Under Load.....	161
Log 9 - Ship It Like You Show Up.....	164
About the Author.....	167
Acknowledgments	168
Embracing the Tools That Accelerate the Work	169

Prologue - Built for More

Before the titles. Before the roadmap, the metrics, or the barbell plates neatly stacked on a rack. There was the mission.

I've worn uniforms and hoodies, sat in war rooms and boardrooms. I've served with soldiers and shipped with engineers. And the one thing that always stood out — the one thing that always worked — was giving a damn. About the work. About the people. About the outcome.

This book isn't a memoir. It's not a manual. It's a field guide for anyone trying to build something that lasts — whether that's a body, a product, a company, or a life with integrity.

You won't find shortcuts here. Just reps. Principles. Stories from the weight room and the war room — forged in pressure, refined in reflection.

Because real strength doesn't come from just lifting the weight. It comes from knowing *why* you're lifting it. Now, let's get to work.

- Mike

1

Mission Before Metrics

The Mission Is the Magnet

Before I ever built products, I served in the U.S. Army as an Airborne intelligence sergeant. That experience — of working in service of something larger than myself, of making decisions under pressure with lives on the line — shaped everything that came after. I learned early that **mission comes first**. Not ego. Not recognition. Mission.

That mindset followed me from the military to cybersecurity, and into leadership roles where the stakes changed but the values stayed the same. Whether it was securing critical systems, helping build Elastic Security, or coaching in the gym, the goal has always been the same: **real strength is lifting others**.

One of the proudest chapters of my product career was building Endgame. We entered a brutally competitive market — going head-to-head with massive players like missionbuilt.io

McAfee and CrowdStrike — and carved out real ground. Not because we had more money or brand recognition, but because we had something harder to copy: **a clear mission.** Protect high-value targets from nation-state level attacks. That focus, and the small, fierce team who rallied behind it, made all the difference.

At the heart of it was Nate Fick, a Marine officer turned tech CEO, and later the U.S. Ambassador for Cyberspace and Digital Policy. He led with conviction, clarity, and a deep respect for the mission. In all-hands meetings, Nate would remind us that we were an "elevator asset company" — that if the building burned down, the most important assets could still fit in the elevator. It wasn't the code or the tools. It was the people — the ones who understood the user's mission and had the passion to serve it.

That idea stuck with me. Nate's example reinforced what I learned in uniform: **The success is the user's success. Your mission is their mission.**

Metrics are the outcome of making your user successful. Yes, we need to measure them. But they are the *result* of serving the mission — not the *reason* for it.

There's a moment in every product meeting when the question slides in like it always does:

"How will we measure success?"

It's a good question — just not always a good *first* question.

In lifting, it's the same story. People chase PRs every week like the number on the barbell is the whole point. Add five pounds. Hit record. Post the clip. Repeat.

But metrics without mission? That's just noise. Pressure with no direction. Goals with no guts.

"The weight on the bar isn't the goal — it's the evidence of progress, not the destination."

We've all seen what happens when this mindset takes over. It's not a failure of talent — it's a failure of alignment. *Cyberpunk 2077* didn't initially flop because the devs didn't care — far from it. The development team poured years into building something ambitious. But the pressure to hit a holiday launch window — a decision made at the executive level — overrode the mission of delivering a complete, polished experience. The result was a rocky release, millions in refunds, a reputational hit, and a stock crash.

To their credit, the team stuck with it. Years later, after patches and a reimagined DLC, the game has earned back much of the trust it lost — a testament to what happens when talented people are finally given the space to do the work right.

And we've seen what it looks like to protect the mission — even when it means stepping back. Simone Biles did exactly that in front of the entire world at the 2021 Olympics. Under unimaginable pressure, she chose long-term purpose over short-term performance. She knew something was off, and she honored that instinct.

"I have to focus on my mental health... if you don't, then you're not going to enjoy your score and you're not gonna succeed as much as you want to."

- Simone Biles

Her move wasn't retreat — it was leadership. And in time, she returned to competition stronger, on her terms, and more respected than ever. A different kind of comeback — one powered by mission, not metrics.

Success isn't about hitting every metric. It's about refusing to lose yourself trying.

More Than Just Good Intentions

Mission-driven isn't a poster in the break room or a bullet in a pitch deck. It's how you move — how you decide, how you show up when it's hard.

In a world that celebrates velocity, mission is quiet. But that doesn't make it weak. Mission gives you *clarity*

when things get blurry and stamina when things get hard.

Take Patagonia. They once ran a campaign telling customers *not* to buy their jacket unless they truly needed it. Why? Because their values mattered more than their quarterly revenue.

Or SpaceX — aiming at goals that span decades, not quarters. It's not about idolizing companies. It's about recognizing what real mission-alignment looks like when the stakes are high and the timelines are long.

And it pays off. Research shows that people who believe in the purpose behind their work stay longer, burn out less, and deliver more.

When Metrics Eclipse Meaning

Let's be clear: metrics matter. But only when they serve the mission — not when they *become* it.

Here's where teams lose the plot:

- They ship fast instead of shipping right.
- They chase signups instead of learning why users leave.
- They brag about launches and ignore long-term usage.

When dates drive development, teams start cutting corners. Feedback loops close. Energy fades. You build momentum toward a number — not toward value.

In the gym, this is ego lifting. In product, it's just as risky. It looks like burned-out engineers, brittle systems, and growth that collapses the second you stop pushing it uphill.

The Fulfillment Flywheel (Powered by Purpose)

There's a better model. One that's as relevant in combat as it is in code — or in the squat rack.

It's called the **OODA Loop**: Observe, Orient, Decide, Act. Developed for fighter pilots. Adopted by startups. Powered by clarity.

But here's the catch — without a clear mission, the whole loop spins out.

OODA Stage	With Mission-Driven Focus	Without It
Observe	You know what matters to watch	You collect everything, drowning in noise
Orient	Purpose helps filter & frame inputs	Metrics get over-prioritized, lose big picture

OODA Stage	With Mission-Driven Focus	Without It
Decide	Mission becomes a north star for action	Risk of chasing vanity wins or short-term gains
Act	Execution has energy and resolve	Actions may be misaligned or half-hearted

In lifting, it's trusting the plan instead of maxing out because you feel good that day. In product, it's waiting to ship because your users aren't ready — even if your OKRs are.

Mission turns chaos into clarity. It makes every rep count. Every release matter. Every decision directional.

This is the real flywheel of fulfillment:

Mission fuels clarity.

Clarity powers resilience.

Resilience drives real progress.

And real progress reinforces the mission.

That's the through-line. That's what keeps us going.

Metrics follow. But the mission leads.

The Drift

At first, the metric is a mirror — it reflects the mission. Clean, focused, and true.

But over time, something shifts. Quietly. Gradually. The mirror warps. And without noticing, we begin steering toward the reflection instead of the road.

This is the Drift.

It doesn't announce itself. It rarely arrives with bad intent. It begins with a goal:

- Launch the feature by Q4.
- Hit 405 on deadlift.
- Increase MAUs by 20%.

Reasonable. Measurable. Actionable. These are the numbers we hold up as evidence that we're making progress — and for a while, they are.

But then: the goal becomes the game.

In product, we've seen it time and time again. Shipping becomes more important than solving. Teams crunch to meet a date set quarters ago, long after the user problem has evolved. Metrics chase headlines. Investors

want news. Executives want motion. The dashboard glows green while user trust fades red.

Think of Boeing. In the race to beat Airbus, they needed the 737 Max on runways — fast. Shareholder pressure mounted. Deadlines became immovable. On paper, the metrics looked great: deliveries met, costs controlled. But beneath the numbers, safety systems were skipped, warnings dismissed.

Two crashes. Hundreds of lives lost. The drift wasn't just technical — it was cultural. Speed overtook scrutiny. The metric overtook the mission.

In strength, the signs are physical — and brutal. Hafthor Björnsson, one of the strongest men in history, returned to powerlifting after two years of boxing. He set his sights on breaking the raw total world record. Big lifts were stacking fast. But recovery lagged behind. He felt the warning signs — tightness, fatigue — but kept pushing.

I wasn't recovering fully between sessions... not listening to my body, which is silly.

Then came the snap. Attempting a 556-pound bench press, his pec tore clean off the bone.

The drift had found him too: progress misaligned with process, recovery sacrificed to reach a number that no

longer served the mission. The result was pain, delay, and a lesson carved in scar tissue.

The damage isn't just physical. It's psychological. Because when we treat metrics as the mission, missing them feels like failure. And so we hide the truth, inflate success, or worst of all — stop trying.

The Drift corrodes not through force, but through **inversion**. It flips process into performance. It turns care into compliance.

And it burns people out.

I've worked with brilliant people who could have built anything — but they left, not because they failed, but because the system stopped valuing *why* they showed up in the first place. When we reduce contribution to a dashboard, we forget the soul behind the keyboard.

This isn't just anecdotal. Research shows that mission-driven employees are significantly more loyal.

A LinkedIn survey found that employees motivated by mission were **54% more likely to stay** with their company for five or more years.

Another study revealed that companies engaging employees in purpose-driven programs saw a **52% lower turnover** among newer employees.

So what's the antidote?

It's not to ignore metrics. It's to anchor them. To use them as signal, not steering.

To build systems that reinforce *why* we do the work, not just how fast we do it.

Because the real goal is not a launch date or a deadlift.

The real goal is built through the reps, not measured by them.

Repetition with Intention

You don't get strong by lifting heavy once.

You get strong by showing up again. And again. And again.

But if you do the same thing forever, you don't get stronger — you get stuck.

That's the tension of progress: it demands ritual, but it punishes repetition without variation.

Whether in the gym or in product, growth comes through iteration — small, deliberate cycles of effort. You log your training. You tweak your form. You try a new tempo. In software, you ship. You measure. You talk to the user.

You try again. The process isn't glamorous — but it's generative.

Rituals compound. Rules confine.

A rule says "do this."

A ritual says "do this because it matters."

One is brittle. The other bends with you.

Agile, for example, isn't magic. It's just a set of rituals.

Daily standups, retros, demos, sprints — all meant to create rhythm and reflection. But rituals only work when they're **anchored in meaning and adapted to context**. If your team treats retros like checkbox theater, then you're not iterating — you're just looping. And if your standup sounds like "I did stuff, I'll do stuff, no blockers," you've got a ritual without reason.

The same failure shows up in the gym. People find a powerlifting program online and follow it to the letter, even when it doesn't fit their recovery, their age, their job, their life. They wonder why they plateau — or get hurt. It's because they forgot the principle beneath the plan.

That's where most systems fail: not because the framework is flawed, but because the **user is forgotten**. They're treated like an input to a method, instead of the reason for its existence.

Let's be honest: there's no shortage of books telling you how to do things.

And yes — here we are, writing another one.

But this isn't a blueprint. It's a **philosophy**.

The implementation is on you.

What works for one team, one lifter, one body, won't work for another. That's not failure — that's reality.

Stefi Cohen didn't become a record-breaking lifter by blindly following a single powerlifting template. Her training evolved through experimentation: strength phases, hypertrophy cycles, hybrid athletic blocks. She didn't just lift heavy — she studied her feedback, varied her stimuli, and rewrote her rituals to fit her mission.

Progress didn't come from rigidity. It came from rhythm and reinvention.

In product, that same spirit of experimentation lives in hack weeks — short, sacred breaks from routine where teams can stretch in new directions.

Atlassian, Spotify, and Elastic have all embraced this. At Elastic, hack weeks became a celebrated ritual — not just for innovation, but for joy. Engineers got to break free from roadmap gravity and chase ideas that didn't need to justify themselves in Jira. Some of Elastic Security's most creative features — including internal

workflow improvements, experimental visualizations, and early prototypes of user-requested tooling — were born not in sprint planning, but in that carved-out chaos.

Sometimes, the best way to realign with the mission is to deliberately step outside it.

Rituals work when they're shaped by the user, not imposed on them.

They're tools — not commandments.

And when they're working, you feel it: not just in output, but in outlook. A team that trusts its rituals doesn't need micromanagement. A lifter that trusts their program doesn't need motivation hacks. The rhythm carries you. The meaning sustains you.

But rituals aren't static.

They must flex to your fatigue. They must bend to your bandwidth. They must evolve — not just to avoid boredom, but to stay honest with the mission.

Same input, same outcome. If the goal has changed, so must the reps.

That's what separates the lifter who grows from the one who stalls.

That's what separates the product team that adapts from the one that burns out.

Because if the mission is the magnet —**rituals are the rails.**

They don't tell you where to go.

They keep you from sliding off the path while you find it.

Rituals give you direction.

But progress? That comes from the work itself — the grind, the reps, the effort no one applauds.

2

Built Through Reps

The Myth of Overnight Success

We've all heard the stories. The product that "took off overnight." The lifter who casually pulls four plates like they've always been able to. The founder in a garage who changes the world with a single keynote.

It's tempting to believe that mastery happens like that — in a flash.

But that's not really how it works.

Take the iPhone. The 2007 reveal made it feel like it dropped from the sky fully formed. But behind that moment were years of trial and error. Touchscreens that didn't quite work. Software that crashed mid-demo. A mountain of prototypes that never saw the light of day. The final product wasn't a stroke of brilliance — it was the result of relentless iteration.

Same with Instagram. What looked like an overnight success was actually the second version of a too-close-to-Foursquare check-in app called Burbn. The team just kept listening, trimming, trying again. One update at a time.

And that big deadlift on your feed? Probably not magic. More likely: a hundred quiet sessions, some of them rough. A lot of small choices to show up, tweak form, trust the program.

We love the idea of sparks — fast wins, big leaps, sudden breakthroughs. But progress usually doesn't feel like that.

Most of the time, it feels a lot more like repetition.

The engineer fixing the same piece of code — again. The founder rewriting their pitch for the fifth time. The lifter doing the same warm-up cues every session, no matter the weight.

It's not flashy. But it adds up.

That's part of what makes the first few months — or the first year — so exhilarating. In the gym, you make gains almost every week. In a startup, your product evolves daily. Everything feels fast, and the feedback is loud.

But then you hit the plateau.

The easy wins dry up. Strength doesn't come as quickly. Users get harder to surprise. Suddenly, it's not about chasing sparks anymore — it's about showing up and pushing through.

This is where the real work begins.

In lifting, that means grinding through the middle — not maxing out, but mastering the basics under load. In product, it's navigating the shift from scrappy innovation to enterprise-grade reliability. Less fanfare, more focus.

That's the part people don't always talk about — and the part that actually defines mastery.

It's not just about adding more weight or shipping more features.

It's about learning to *hone your form*.

In lifting, that might mean dialing in your brace, fixing a subtle shift in your squat, or finally feeling your lats engage in a deadlift.

In product, it's refining an initiative until it truly solves the right problem — not just adds to the roadmap.

You start to realize: Reps aren't just about volume. They're about attention.

That's what makes progress sustainable.

And that's what *Mission Built* is really about — **building better products, one rep at a time.**

Repetition Is Not Redundancy

In Section 1, we looked at the myth of overnight success — how real strength, in the gym or in product, is built through showing up again and again. But not all reps are created equal. The real magic isn't just in the repetition — it's in how you use it.

Not all reps count the same.

Anyone who's coasted through a gym session, just moving weight, knows the difference. One set burns time. Another builds awareness, precision, and strength — even if the numbers don't change.

The same is true in product.

Repetition isn't redundant when it's intentional. When it's used to test assumptions, refine interfaces, tune performance, or build something just a little closer to what users actually need. That's the kind of rep that moves things forward.

But repetition alone isn't enough — **variation is what makes repetition transformative.**

As Dr. Mike Israetel, co-founder of Renaissance Periodization, teaches:

"Training works best when structured into intentional blocks that emphasize different goals. In hypertrophy-focused blocks, lifters increase volume over time — guided by principles like Minimum Effective Volume (MEV), Maximum Adaptive Volume (MAV), and Maximum Recoverable Volume (MRV). This structured overload builds muscle while preserving recovery, setting the stage for later strength expression. The concept isn't just about working hard — it's about working smart, within clear, adaptive boundaries."

Source: <https://drmikeisraetel.com/dr-mike-israetel-mv-mev-mav-mrv-explained/>

In training, we use blocks for different goals: hypertrophy to build muscle, strength to build expression, and deloads to recover and grow. We manipulate movement, load, and tempo to avoid stagnation. A paused bench press, a deficit deadlift, or chains added to a bar — each variation targets a different adaptation, even if the base movement stays the same.

Product is no different.

You might run a block of sprints to achieve a specific outcome — refining onboarding, improving performance, or tackling tech debt. Then shift into a new block with a new goal. Like training, it takes enough intentionality to complete a cycle, maybe one, two, or three sprints, before varying the focus. Progress comes not from flailing in every direction, but from **focused variation over time.**

And just like strength doesn't grow without tension, product insight doesn't grow without diverse input.

You can't just loop with your own team forever. Sometimes you have to stretch — to talk to sales, to customers, to analysts, to skeptics. Even when it's uncomfortable. Especially when it is.

This is where The Medici Group gets it right: innovation happens when diverse perspectives collide.

As Frans Johansson puts it in The Medici Effect:

"When you step into an intersection of fields, disciplines, or cultures, you can combine existing concepts into a large number of extraordinary new ideas."

In the gym, that might mean lifting with people stronger or different than you. In product, it means bringing in

voices from outside your echo chamber — support calls, sales objections, the person who almost churned.

Reps alone build endurance.

Smart variation builds power.

This is how you break through the plateau — not by abandoning the reps, but by evolving them.

When the Spark Fades

The first reps are easy — not physically, but emotionally. You're fired up. Everything's new. Progress is loud and obvious.

But what happens after the rush?

In lifting, it's the long middle. The early PRs stop coming. Your form stalls. You show up, grind through the same sets, and wonder if you're actually moving forward. You're not always chasing your one-rep max. And you shouldn't be. Strength isn't built by living at your limit — it's built in the space between peaks, when you train with intention and recover with discipline.

In product, it's the same. You can't always be doing the flashy, innovative thing. Sometimes the most important work is foundational — fixing backend debt, improving

accessibility, tightening up performance. The kind of work that creates capacity for brilliance later.

And for many, this is where the wheels come off.

Because motivation — that spark — is unreliable. It's not designed for the long haul. And it doesn't care about your goals.

That's why systems matter more than sparks.

Systems are how you keep showing up when the dopamine dies down. Morning routines. Logbooks. Standups. Progress reviews. They don't need to be rigid. But they do need to be real. Reps don't get done by accident.

You don't need hype — you need structure.

In training, that structure might be a coach, a program, a calendar alert that says “get under the bar.” In product, it might be a rhythm of sprint planning, async demos, or check-ins with customers. Externalized accountability is often the only thing that keeps momentum moving.

James Clear — author of the bestselling book *Atomic Habits*, known for his work on behavior change and habit formation — wrote:

"You do not rise to the level of your goals. You fall to the level of your systems."

The work still has to be done. But when your environment supports your actions, it gets done more often.

And here's the quiet truth:

The people who make the biggest progress aren't usually the most intense — they're the most consistent.

They build when no one's watching.

They train when it's not fun.

They keep caring — even when the spark is gone.

But what if you need to *find* the spark again?

One of the fastest ways to reignite your drive is to reconnect with the people you're building for. Talk to your users — especially the ones who aren't shouting. You might think the work you're doing isn't flashy, but somewhere out there, someone is craving exactly what you're building.

Stability. Accessibility. Visibility. These aren't buzzwords — they're lifelines for users with real problems. Find them. Listen to them. Let them lift you up.

Because nothing recharges momentum like hearing someone say: “*This made my day better.*”

And that brings us to the next section: the power of the quiet reps — the multiplier of boring work.

The Multiplier of Boring Work

There’s a kind of work that doesn’t make headlines. No one posts their warm-up sets. No one celebrates shaving 100ms off load time.

But that’s the work that wins.

For every PR pulled in competition, there are hundreds of days of grinding behind it — submaximal sets, long pauses, light reps, mental resets. The same is true in product. Every effortless-looking release rests on a foundation of something much deeper: months of planning, iteration, bug-fixing, and late-night Slack threads.

In lifting, it’s the mobility work you do alone at 6 a.m. The back-off sets you don’t skip. The deload week you take seriously. It doesn’t look impressive — but it makes everything else possible.

In product, it's building out role-based access controls — not because it's exciting, but because your biggest customers expect it. It's mapping audit logs across services so your platform isn't a compliance risk anymore. It's the 10th conversation with a user about the same rough edge in the UX. These aren't "big bets," but they're the reason your big bets land.

That's what boring work does: it compounds.

Each rep you don't skip, each ticket you don't shortcut, each problem you refine instead of avoid — it stacks. Quietly. Relentlessly. And over time, it becomes your edge.

You don't need to go viral. You need to be trusted.

And trust is built in the boring work.

The warm-up that prevents injury.
The small fix that prevents churn.
The five-second improvement that gives a user five minutes back.

This is what separates the strong from the strong enough.

It's not what you do once.
It's what you do without applause.

You do it for the growth. For the discipline. For the user whose day you quietly made better. Not for the accolades.

That's what separates long-term success from short-term effort — not glory, but the passion to do the work for its own sake. The features and the gains? They're just symptoms. What matters is the mission that fuels them.

That's why I'm so passionate about product management — and about lifting. Because the best PMs and the best lifters don't just show up for themselves. They show up for the team, for the user, for their own growth — not in a selfish way, but in a way that elevates everything and everyone around them.

And if you've made it this far — through the reps, the plateaus, the quiet work — you already know:

This isn't just about shipping or lifting.

It's about becoming the kind of person, or the kind of team, that keeps showing up.

That's the real win. And that's where we end this chapter — not at the peak, but at the foundation.

The Work Becomes the Win

At some point, the reps stop being something you *have* to do. They just become something you *do*.

You stop chasing motivation and start trusting momentum.

You stop asking when it gets easy and start asking how to keep showing up.

And that's the shift — not just in the gym or in your sprint board, but in your mindset. You lift because you care.

You build because it matters. Because you love it.

It's not about PRs or product launches. It's about what they represent: The hours you logged. The patterns you learned. The people you helped.

That's what it means to be *mission built*.

You're not doing it for the spotlight.

You're doing it because you've seen the power of the process — and you're not walking away from it.

Progress doesn't shout — it stacks.

One quiet rep at a time.

3

Rituals Over Rules

The Ritual Is the Rail

Rituals keep us on track — not by forcing us forward, but by guiding our momentum. Like rails under a train, they don't power the engine, but they make sure we're headed in the right direction. They keep drift from becoming derailment.

In the gym, rituals aren't written on the whiteboard, but everyone knows them. The way you chalk your hands before a heavy lift. The nod from your training partner. The slap on the back before a max attempt. These moments don't make you stronger by themselves — but they signal that it's time to lock in. They tune your mind and body to the work ahead.

In product, rituals shape how we build — and why. A startup might begin with a single weekly sync, focused on shipping a differentiated capability that makes a mark on the market. That urgency, that clarity of purpose, fuels early momentum. But as the company grows, the

rituals evolve. Roadmap reviews become more structured. Standups get tighter. You start shipping the “boring” features — the ones that don’t win awards, but win hearts. Role-based access control. Data retention workflows. RBAC and audit logging won’t make the cover of TechCrunch, but they’re what your biggest customers need to trust you.

These shifts aren’t a loss of agility — they’re a deepening of commitment. The rituals grow because the mission does.

The difference between a rule and a ritual is intention. Rules are imposed. Rituals are earned. A rule says “follow this or else.” A ritual says “this is how we move with purpose.” When teams stop questioning *why* they follow a process, it calcifies into dogma. But when the process stays tied to the mission, it adapts. It grows. It stays human.

Elite lifters know this instinctively. Even as their programming changes — higher volume, lower intensity, or a shift from competition prep to recovery — their rituals stay intact. The warmup flow, the music choice, the way they approach the bar. These things evolve, but they never disappear. They guide the body into readiness. They keep the lifter connected to something deeper than the day’s numbers.

And that's the point.

You don't need more rules. You need rails that carry you toward the mission — and flex with the turns ahead.

Because if the mission is the magnet, the ritual is the rail — not to restrict your path, but to keep you from drifting when the pressure builds.

Listen to Your Signals

Rituals only work if you're paying attention.

You can't run the same program forever. Not in the gym. Not in product. Not in life. Progress demands feedback. And feedback starts with listening.

Lifters learn this early. You might show up ready to deadlift heavy — but your grip feels off, your back's tight, your CNS just isn't firing.

(**CNS**: Central Nervous System — the part of your body responsible for strength output, coordination, and neural drive. If it's fatigued, you'll feel it, even if your muscles are technically rested.)

That's not failure. That's information. A smart lifter doesn't abandon the workout — they adjust. They keep the ritual, shift the intensity. Maybe you pause at 70%,

maybe you pivot to accessories. Listening doesn't make you weaker. It keeps you in the game longer.

In product teams, the same truth holds: rituals without awareness become liabilities.

You can run the ceremonies — sprint planning, retros, standups — but if you're ignoring signals from your team or your users, you're performing process theater.

Burnout doesn't show up in Jira. Disengagement doesn't flash red in a dashboard. You feel it in the delay before someone unmutes. In the tension after a roadmap shift. In the quiet attrition of both teammates and customers.

Nowhere is this more visible than in the game industry's long-standing reliance on **crunch time**. Late-stage death marches, where teams work 60-, 70-, 80-hour weeks to hit a ship date. Executives cite passion. Teams call it what it is: avoidable. The rituals of "just one more sprint," of all-hands war rooms, of praise for pulling all-nighters — they're treated as signs of commitment, when they're actually signals of failure.

Failure to listen. To plan. To build sustainable systems.

Failure to treat the team as human — not just headcount.

This is where product management has a second, often overlooked role.

Yes, product is the voice of the user inside the development team — but product is also the shield of the team *against* the wrong voices from above. A good PM doesn't just absorb pressure from the top and pass it down. A good PM pushes back. Uses data to say no to date-driven development. Advocates for pacing, not panic. Protects the team's ability to think, breathe, and build well — even when the deadline is loud.

Because our goal isn't to meet an investor's timeline or an executive's forecast.

It's to build the best product to solve our user's mission.

And the only way to do that — sustainably, meaningfully, and well — is to listen.

To your body.

To your team.

To your users.

To the mission.

Change the Pattern, Not the Practice

When things stop working, you don't abandon the ritual — you adapt it.

Progress isn't a straight line. It's a cycle of push, plateau, pivot. And when you hit that plateau — when the usual ritual no longer delivers results — it's not a sign to quit. It's a sign to evolve.

In the gym, this happens all the time. You run a successful squat cycle, adding five pounds a week like clockwork. But then you stall. Your knees cave, your speed slows, and the bar starts winning. The answer isn't to scrap the movement. It's to change the pattern. Maybe you shift to pause squats, change your stance, or drop the volume to focus on recovery. The ritual — showing up and squatting — stays. The shape of it changes.

The same holds true in product.

The standup that once helped your team sync becomes a box-checking chore. The planning meeting that used to set priorities now spirals into status updates. That doesn't mean you stop planning. It means you shift how you plan. You move from live check-ins to async threads. You replace calendar fatigue with focused, flexible rituals that match your team's needs — and their current phase of growth.

Rituals that don't evolve become rules. And rules, as we've seen, break people.

This is especially true in globally distributed teams. At Elastic, our workforce spans time zones and continents. We had to learn — early — that rituals built for co-located teams don't translate. Sync meetings at 10 a.m. in California are 7 p.m. in Berlin, and 2:30 a.m. in Sydney. So we changed the pattern.

We moved toward asynchronous rhythms:

- Key decisions live in documents, not meetings.
- Slack threads replace sidebars.
- Recorded meetings include transcripts and tagged callouts.
- Big calls are followed by quiet time — space for global teammates to reflect, respond, and contribute.

We didn't stop collaborating. We just restructured how and when collaboration happens. The ritual of cross-team communication stayed. Its shape changed. And it made space for more voices, not fewer.

What matters isn't how you do it.

What matters is that you keep showing up with purpose.

The gym teaches you this without words.
You hit a wall. You adjust. You don't abandon the work —
you change the pattern.

Because in the end, progress comes from consistent
effort, not perfect conditions.

That's the lesson: rituals serve the mission. When they
stop serving, reshape them.

Don't confuse rigidity for discipline.

Don't let the form matter more than the function.

(And please — don't treat your favorite product
management book like doctrine. Those frameworks
weren't written for your team. Your mission was.)

Rituals aren't constraints — they're commitments. But
only if we let them evolve. The strongest systems, the
longest-lasting teams, the most resilient lifters — all
share this in common: they don't cling to routine for
routine's sake. They adapt. With intention. With
feedback. With mission in mind.

4

Feedback Is A Superpower

Cues, Not Critiques

My mentor Nate Fick used to talk about the power of **laser focus**. Not just attention — precision. That phrase gets thrown around a lot, but he meant it in the truest sense.

Laser focus isn't about doing one thing. It's about doing the **right** thing — the move that creates the most impact. The shift that makes everything else click into place.

I saw that principle in action during my early days at Endgame.

When I joined, we had four different products — all in different markets, serving different users. Built with a small team. Sold with an even smaller one. The problem wasn't that the ideas were bad. They weren't. In fact, each had real potential.

But potential doesn't scale without focus.

Nate made a tough call. He chose to end-of-life three of the products — even though they were generating short-term revenue — and put every ounce of our talent behind a single one.

It was a risk. But it paid off.

We built a best-in-class endpoint protection product in one of the most competitive markets in cybersecurity. Not because we chased everything. But because we aligned everything to a singular mission.

That's what laser focus looks like in the real world. And feedback should work the same way.

When you're under pressure — under the bar, under deadline — you don't need a list of everything that's off. You need one cue. One phrase that slices through the noise and sticks when it counts.

Point the Flashlight

Like in baseball.

Coaches used to flood hitters with advice mid-swing. “Keep your hands back.” “Level the bat.” “Don’t bail out.” All technically true. None actually helpful in the moment.

Then came this:

Picture the knob of the bat as a flashlight. In the "flashlight position," you're aiming that light beam right at the catcher's face. This cue helps you set the correct bat angle — slightly upward, with the knob directed toward the target. It's called the flashlight position because you're holding and pointing the bat the same way you'd shine a flashlight in someone's eyes.

“*Flashlight!*”

It's weird. It's vivid. And it works.

The batter stops overthinking and starts moving with intention. The hips rotate. The swing tightens. Power follows.

That's the magic of laser-focused feedback — not more, just clearer.

Two Words That Changed the Lift

The same thing happens under the bar.

Stefi Cohen is one of the most accomplished powerlifters on the planet. But even the best get stuck. During a heavy squat cycle, she was losing depth and collapsing forward — the kind of breakdown that could invite a checklist of mechanical critiques.

Her coach didn't do that. He gave her one cue:

“Knees out.”

Two words.

Her glutes fired. Her hips opened. Her position locked in. No overcorrection. No overthinking. Just a precise adjustment that reconnected the movement.

That's what real feedback does. It doesn't flood you with everything that's wrong.

It's a spotlight, not a floodlight — aimed at what matters most, right now.

From Backlog to Breakthrough

Similarly, in product development, we're inundated with feedback.

Users, developers, sales teams, executives — everyone has ideas, and most of them are shared with genuine excitement. Everyone wants the product to be better.

But without focus, we risk becoming a jack of all trades and a master of none. Chasing every suggestion means delivering none of them well.

The best product teams don't just collect feedback — they hone it. They find the thread, the theme, the one insight that can generate the most impact with the least investment.

Because here's the truth:

In software, we can build almost anything — given infinite time and budget.

But for some reason, executive teams rarely offer those.

So it takes real leadership to sift through the noise, spot what matters most, and act.

Take an example from Dropbox.

Dropbox had a problem most product teams would recognize. Users were signing up... but not sticking around. Specifically, they were dropping off before installing the desktop client — the core part of the experience that made Dropbox more than just another web app.

The team started spinning on solutions. Maybe the onboarding flow needed to be redesigned. Maybe they needed more education, more nudges, more screens. The backlog of ideas kept growing.

Then a PM said something simple:

| “You’re treating the CTA like a destination, not a trigger.”

That one line cut through the noise.

They changed the button copy from “Next” to “Install Dropbox.”

They added a subtle animation to guide the user forward.

That was it.

And conversion went up.

No massive rebuild. No big campaign. Just a focused shift — one that made it crystal clear what the user needed to do next.

It wasn’t about the volume of feedback. It was about knowing which piece would actually move the needle.

| *That’s product leadership: listening broadly, then acting precisely.*

The Flashlight, Not the Floodlight

The same principle applies across every domain — lifting, sport, software.

You don't have infinite time to build a product.
You don't have infinite weights to lift and sculpt the perfect physique.
You can't act on every idea, every error, every impulse.

So you focus.

Whether you're adjusting your squat, refining your swing, or unblocking a user journey — the best feedback doesn't overwhelm. It illuminates.

Not a floodlight that blinds.

Not a laser that burns.

Just a flashlight, steady and clear, aimed at what matters most right now.

That's how progress happens. Not with noise — with precision.

One cue, one rep at a time.

Make Feedback a Ritual

Feedback isn't something you give once a quarter. It's not a Slack comment or a surprise at review time. It's a rhythm. A rep.

If cues are the spark, rituals are the circuit — they keep the current flowing. The strongest teams and the strongest lifters don't just react to feedback when it shows up. They **build it in**. They expect it. They practice it. They normalize it.

In lifting, you log your sets. You reflect on what moved well, what felt off, what needs attention tomorrow. Feedback isn't an event — it's a ritual. Same goes for product.

And if you're a product manager, here's the simple truth:

If you're not talking to users, you are failing at your job.

Not falling behind. Not missing a step.

Failing.

User feedback isn't a bonus. It's not a "nice to have." It's the core ritual of the role. If you're not making regular space to listen — directly, personally — then you're flying blind.

You're not a scrum master.

You're not a backlog babysitter.

You are the voice of the user.

And here's what that really means: the user is not a secondary part of your job. They're not a "requirement" that emerges after the product is scoped. They are not a symptom of your development process.

*The user — and their mission — is **why** you have a job.*

That's the point of all of this.

The research, the roadmaps, the revenue — all of it is an outcome of solving the user's problem. Not the other way around.

Too many teams fall into the trap of building for elegance, or scale, or conversion — and forget to ask the most important question: **Does this move the mission forward?**

That's your job as a PM.

To hold that line.

To make feedback a ritual — not just to improve the product, but to stay anchored to the reason it exists.

Because metrics can drift. Roadmaps can get noisy. But the mission stays clear — if you keep listening.

Slack: Feedback as the Product

Slack didn't start out as a product idea. It started as an internal tool. The team at Tiny Speck built it for themselves while working on a multiplayer game called *Glitch*.

The game didn't succeed. But the internal chat tool did — because the team had built it by listening to their own pain.

They weren't running customer interviews or user panels. They were simply using the tool every day — and treating their own experience as data. The engineers, designers, writers, and founders were all giving and receiving feedback constantly. What was slow? What was confusing? What needed to work better?

The users were *them*. The feedback was *constant*. And the ritual was built into the way they worked.

They didn't treat those observations as complaints. They treated them as cues — small signals that, over time, shaped something far more useful than they had originally intended.

So when *Glitch* shut down, the pivot to Slack wasn't a Hail Mary. It was a progression. The team had already spent years listening to what they needed most — and building it.

Ritualized feedback didn't just improve the product. It revealed the product they should have been building all along.

Slack didn't become Slack by guessing what users wanted.

It became Slack by listening to internal signals — consistently, honestly, and early.

Elastic: Customer Zero and Design Partner Rituals

We've seen the same thing at Elastic.

Our internal information security team — “customer zero” — is often the first to use upcoming Elastic Security features. They’re not a test group. They’re real users with real problems. When something’s unclear, slow, or missing, they tell us — early and often. And because they sit just a few channels away, we can respond quickly and refine before anything reaches broader release.

It’s not just internal, either. Some of our most pivotal feedback comes from design partners — external users who’ve volunteered to be on the front lines with us. They opt into early access, knowing the feature might be raw, knowing it might break — because they care about

shaping something real. Something that works not just in a demo, but in a live environment.

These aren't one-offs.

They're not spontaneous bursts of goodwill.

They're planned rituals — structured partnerships built into how we deliver great products.

Embedded loops that make the product stronger long before it ever reaches general availability.

Ritualized feedback isn't about checking a box.

It's about building alongside the people who **share the mission**.

Relationship-Level Listening

Collecting feedback is the easy part.

Like in any relationship, getting someone to tell you what's on their mind is just the start.

What really matters is what you do next.

Do you listen? Do you actually hear them?
Or do you nod, say "totally," and go back to your plan?

It's the same in product. And in training.

You can run all the standups, retros, interviews, and training logs you want — but if you’re not listening with intent, you’re just gathering noise. You’re not growing.

Feedback is only useful if it lands.

And that means learning how to listen — not just politely, but purposefully.

Listening Is a Lift

You’ve built the ritual. You’re getting the feedback. Now comes the hard part: listening.

Not passively. Not performatively. But actually listening — in a way that helps you grow.

And here’s the first thing to understand:

Being user-obsessed doesn’t mean the user is always right.

Sometimes users give you feedback in the form of a solution.

“Add a new button.”

“Change this flow.”

“Make it more like what Company X does.”

If you chase those suggestions blindly, you're not listening — you're outsourcing product design.

The real job of a product manager is to listen deeper.
To get past the request and find the root.
Don't treat the feedback as the answer — treat it as a clue.

Your job is to uncover the real problem.

One thing I'm adamant about with my PMs:
We don't hand our engineers a solution.
We bring them a problem worth solving.

That means doing the work. Talking to users. Studying how others have handled it. Mapping the friction. Listening for what's unsaid.

We work with brilliant, creative engineers. Our job isn't to steal the "how."
It's to own the "why."
And collaborate on the "how" — together.

Because feedback isn't always clear. It's not always clean.

It can be noisy, misdirected, emotional, subtle, or even wrong.

And when you're under pressure — in the middle of a hard training cycle or a high-stakes product push — it's

easy to hear what you want to hear.
Or to hear nothing at all.

Listening is a skill. One that separates people who just collect feedback from those who actually **use** it.

It's like lifting: you might record every set and log every number, but if you don't know what the soreness is telling you — if you don't know when to push and when to pause — you're not getting better. You're just getting tired.

Sometimes, the most accurate feedback in a training session isn't in the weight or the timer — it's in how the set **felt**.

How many reps did you leave in the tank?

What was your rate of perceived exertion?

RIR and RPE are qualitative cues — and they only work if the lifter is honest.

Just like product feedback only works if the team is actually listening.

When the Data Lies — or We Lie to Ourselves

Take **Fitbit**.

When they rolled out sleep scores, users were thrilled. Finally, a single number to tell you how well you slept. But the feedback loop backfired. People started optimizing for the score — going to bed earlier, waking up later — and still feeling exhausted.

Why?

Because the metric didn't tell the whole story. It rewarded quantity, not quality. It missed the impact of stress, caffeine, or real rest. And instead of helping people understand their bodies, it created sleep anxiety — people chasing a number instead of listening to what their bodies were actually telling them.

| That's the risk with metrics-as-truth: you end up listening to the wrong signal, just because it's the one that's easy to measure.

And the same thing happens at scale.

Take **Target Canada**.

When Target expanded into Canada, the surface signals looked great: strong brand awareness, excited customers, big opening-week foot traffic. But underneath, stores were suffering. Inventory systems were broken. Shelves were empty. Pricing was

inconsistent. And operational feedback from store teams was ignored or deprioritized.

Leadership focused on the shiny signals — the PR, the launch buzz, the optimism — and missed the messy truths on the ground. They listened to the brand metrics and headlines, not the store managers and logistics teams waving red flags.

Within two years, the entire operation collapsed — costing billions and ending in a full retreat.

*When you only listen to what makes you feel good, you stop seeing what's real.
And in product — or business — that kind of selective hearing is deadly.*

Real Strength Isn't in the Signal — It's in the Interpretation

Feedback isn't rare. It's everywhere.

What's rare is someone strong enough to hear it clearly — and humble enough to act on what it really means.

In lifting, that means paying attention to more than numbers.

In product, it means resisting the urge to react — and learning to respond.

You don't need more metrics.
You need better listening.
Listening that filters noise.
That finds the root.
That says: "*I hear you. Let's fix the real thing.*"

That's what strong teams do.

That's what strong lifters do.

And that's what we do when we care more about truth than comfort.

Strong Feedback, Strong Foundations

We should strive for feedback that's constructive — not combative.

I've learned this over the course of my career.
As a passionate product leader, I've had to walk a line:
How do you bring intensity without intimidation?
How do you make sure your excitement doesn't land as insult?

It's not always easy. Passion can blur the edges.
And in fast-moving teams, feedback can come fast and hot — even when it's meant to help.

But here's what I've come to believe:

When feedback is anchored in empathy and structured around the mission, it becomes a lift — not a weight.

It pushes people forward instead of pinning them down.

These lessons apply well beyond product, and well beyond management.

They show up in how we coach our teammates, how we parent, how we partner, how we grow.

If something matters enough to say out loud, then it matters enough to deliver with care.

Because what you say is only part of it.

How it lands — that's what makes it real.

Strong Feedback Is Often Micro

We tend to think feedback has to be a big moment. A review. A reset. A speech.

But most of the feedback that actually changes people — and changes outcomes — is **micro**.

One cue, one adjustment, one rep at a time.

That's how coaching works on the platform.

That's how great PMs guide a team.

That's how trust is built — and how performance gets better.

In the gym, the best feedback is short and specific:

“Brace before the pull.”

“You’re losing tension at the bottom.”

You don't wait until after the meet to say it — you say it in the warm-up room, in the moment, when it counts.

In product, it's the same.

“This feature’s drifting from the core user need — let’s tighten it.”

“That’s a great insight — let’s surface that earlier in the deck.”

Micro feedback avoids defensiveness. It's less likely to be taken personally.

It's just the next rep, done better.

Share Early, Improve Always

Too many teams wait too long.

They polish. They perfect. They fear showing something unfinished.

But by the time the feedback comes, the team is too attached — or too far gone — to adapt.

The best teams share early.

That's the Medici principle: seek feedback while ideas are still flexible — not fixed.

Let people shape the work before it's locked in.

You can see this principle in action at **Figma**.

From the start, Figma built real-time collaboration into their product — not as a feature, but as a philosophy. Designers could share mockups midstream, collect comments in context, and adjust in the flow.

No long feedback loops. No over-polished reveals. Just open collaboration, one comment at a time.

At **Elastic**, we embraced that same mindset — first in design, then across product teams.

Figma became a shared space for early feedback: PMs sketch rough flows, designers wire up concepts, and teams review together *before* a line of code is written. Mock reviews aren't a gate. They're a ritual.

And like everything else, we've evolved it.

Now we're layering in AI tools like Lovable to explore more ideas faster — iterating on copy, visuals, and layouts in minutes, not weeks. It's not about replacing creative work.

It's about accelerating how quickly we *get to the good ideas* — and how early we can gather feedback that makes those ideas stronger.

And feedback doesn't always need to come through conversation.

Just look at **Duolingo**, one of the most successful learning platforms in the world — and proudly Pittsburgh-built.

They've designed *feedback* directly into the product experience.

From streak celebrations to in-app corrections, every interaction is a tiny nudge:

“*You’re close.*”

“*Try again.*”

“*You’re on track.*”

There's no giant review. No sit-down critique.

Just micro feedback, in real time, helping users improve without ever slowing them down.

*The earlier you share, the better the outcome.
The smaller the feedback, the bigger the impact
— if it comes at the right time.*

Strong Feedback Builds Strong People

Strong feedback doesn't come from shouting.
It comes from someone who's *done the work*.
Someone who's built themselves up — not just to be heard, but to be helpful.

The best coaches don't bark orders.
They observe, they care, and they offer what helps — when it helps.

In product, the same rule applies.

You don't build trust by waiting until things break.

You build it one cue at a time. One adjustment. One shared win.

*Feedback that fuels — not frustrates.
That builds — not breaks.*

Even in our personal lives, the principle holds.
A daily affirmation. A kind word. A check-in after a hard day.

These are all acts of feedback — they tell someone how they're doing, and that you're still with them.

That's how we get stronger. As partners. As teammates. As leaders.

Not through silence. Not through softness.

When it's anchored in empathy and aimed at the mission, feedback stops being friction — and becomes fuel.

5

Progress Isn't Pretty

Mastery Is Repetition with Feedback

Malcolm Gladwell's now-famous "10,000 Hour Rule," popularized in *Outliers*, offers a compelling headline: if you do something for 10,000 hours, you'll become world-class. But what's often forgotten — and what Gladwell himself has acknowledged — is that the number only matters *if those hours are deliberate, structured, and improving*. Not just any reps. The *right* reps.

You see it clearly in cybersecurity.

Penetration tests are our version of reps under pressure — simulated real-world attacks run across teams, infrastructure, and products. We don't go easy. We don't telegraph the punch. Sometimes, we don't even tell the blue team it's happening. Why? Because defenders don't get warning in the real world. If we want to build muscle memory, we need to simulate the actual weight.

That's what "perfect practice" really means — not pristine, flawless action, but high-quality, high-context repetition. As athletes say: *Practice doesn't make perfect. Perfect practice makes improvements.*

It's a mindset that transcends domains. In lifting. In product. In leadership.

The Lifting Lens: Periodization and Progression

Walk into any well-run gym and you won't just see random workouts scribbled on a whiteboard. You'll see *programs*. 5/3/1. Texas Method. Juggernaut. Conjugate. Each offers structure, progression, and intentional overload. They're built on periodization — the principle that you can't (and shouldn't) go hard in the same way every day. Instead, you cycle through intensities, movements, and recovery phases. It's not sexy. But it works.

Few have embodied this more successfully than Chad Wesley Smith, founder of Juggernaut Training Systems. A former world-class powerlifter and coach, Smith built JTS into a globally respected platform by emphasizing not just heavy lifting, but smart lifting — rooted in data, long-term programming, and adaptive feedback. As he puts it:

“Motivation gets you started. Discipline keeps you going. Feedback tells you where to go next.”

— Chad Wesley Smith, founder of Juggernaut Training Systems

I use the JuggernautAI app myself — not because it's flashy, but because it delivers structure. It takes in feedback on how I move, how I recover, how I progress — and adjusts the plan accordingly. Every session has a purpose. It's a coach in your pocket, and a powerful reminder that great outcomes come from smart, consistent reps.

And here's the truth lifters learn the hard way: showing up and doing *something* beats waiting for the perfect plan. A light session with focused movement and intention will outperform a skipped workout every time. Small improvements, made consistently, create compounding strength.

The Product Mirror: Disciplined Iteration at Elastic

When Elastic Security shipped its initial security analytics and SIEM product in 2019, it wasn't a complete answer to the market. It was a start — informed by users, opinionated in structure, and designed to evolve. What made it successful wasn't just the launch. It was the loop: weekly user feedback, signal from real-world

deployments, and disciplined iteration sprints that followed a clear roadmap.

Each product cycle wasn't a shot in the dark. It was a planned lift — part of a broader program of growth. Features like detection rules, case management, and timeline didn't just "appear." They were reps, refined under load. We looked at how users engaged, what didn't click, what needed retraining. We built telemetry into the stack — not to track vanity, but to see where improvement was possible.

Reps With a Purpose

Whether you're logging lifts or logging bugs, the idea is the same: feedback closes the loop. Reps without it just reinforce habits — good or bad. Reps with it? That's where you find progress.

Athletes track tempo, range, and fatigue. Product teams track usage, drop-off, and latency. Neither discipline rewards brute force for long. But both reward those who show up, take notes, and return better.

In the gym, a lifter might swap from high-bar squats to pause squats to address sticking points. In product, a team might switch to dogfooding an unfinished feature to catch friction early. In both, the rep is the mechanism. The lesson comes from how you interpret it.

The Grind That Grows

The reps that change you aren't the glamorous ones.

They're not the PR attempts. Not the launch keynotes. Not the LinkedIn flexes. The reps that build real strength — in the gym and in the product — are the ones nobody sees. The early morning warmups. The late-night bug fixes. The process of showing up, adjusting, refining, repeating.

This is *the grind*. And it's where the gains are made.

In the Gym: Warmups, Not Max Outs

You can spot the novices in any weight room: they go straight for the heaviest lift they can manage. They max out on deadlifts, post it on Instagram, then disappear for weeks.

The ones who stick? They warm up. They move methodically. They hit moderate numbers with perfect form. They log their progress. And they show up again tomorrow.

Warming up isn't a sign of weakness. It's the mark of a seasoned lifter. It's protection. Preparation. And part of

the process. As *Chad Wesley Smith* and countless coaches remind us — progress is built on consistency, not bravado.

In Product: Quiet Wins That Matter

Product teams can fall into the same trap. We celebrate the big splashy features, but the real loyalty comes from the quiet wins — the “minor” UX updates that shave 10 clicks off a daily workflow. The improved loading time. The default that finally makes sense.

We saw this first-hand at Endgame. Yes, we differentiated on high-end protections like in-memory attack prevention. But the reason our champions stayed engaged wasn’t just the cool stuff — it was the continuous improvement in how they used the product. With each release, we delivered smoother experiences and more intuitive workflows. It added up.

The same holds true at Elastic Security. We spent over a year deeply focused on the alert triage experience. We met with analysts. We compiled hundreds of hours of feedback. We made sure everything they needed — context, actionability, next steps — was right at their fingertips. That wasn’t a flashy launch. It was *grinding*. And it paid off.

Not every enhancement gets a press release. But the users notice — and often celebrate — the small, thoughtful changes more than the big ones.

What the Grind Teaches

The grind teaches humility. Discipline. Attention to detail. It teaches you to care about the user who files enhancement tickets, not just the buyer who signs the deal. It teaches you to find progress in the seemingly minor.

In product, it's tackling the long list of enhancement requests. In lifting, it's refining your bar path or ankle mobility. Both require you to step away from the spotlight — and into the discipline of reps.

The Myth of Motivation

The myth is that motivation carries you. The truth? Motivation gets you started. But it won't carry you through the hard sets, the long roadmap, the friction.

What will? Mission. Ritual. Feedback. Community.

We said it in Chapter 1: *their success is your success.* Their mission is your mission. And everything that improves their ability to achieve it — no matter how small — is critical.

The best product managers and lifters don't chase the end state. They chase the process. They *fall in love* with it. And that's where mastery begins.

Look Back to See the Gains

Progress is sneaky. When you're in it every day, it's hard to see.

In lifting, that's why we take progress pictures — not for vanity, but for truth. I've had times where I felt stuck, only to look at side-by-side photos and realize I was growing all along.

Same in product.

One of the best presentations I gave at Endgame was a demo showing the same workflow over multiple years. Each version looked small on its own. But side-by-side? It was night and day. Better design. Better defaults. Better detection. That's the power of consistent improvement.

Change the Phase, Not the Process

Anyone who's followed a good lifting program knows the feeling: what used to work, stops working. You stall. Plateau. Feel stuck.

The worst response? Panic and start over.

The best? Adjust the phase. Keep the process.

In powerlifting, this is where *periodization* becomes personal. You might deload. Shift focus to speed or hypertrophy. Swap bars or change stance. But you don't throw away your structure — you modify it with purpose.

The same is true in product development. You don't abandon the mission when progress slows. You evolve your strategy. You change the shape of the work.

Product Example: Scaling for What's Next

Early in Endgame's life, we took an almost nine-month pause on new features to rearchitect the platform for long-term scale and performance. It was a scary bet, and there were plenty of sleepless nights wondering if it was the right call.

But when we brought the new build to a major deployment — and saw the blue team “win” their live training exercise for the first time — we knew it had paid off. Their expertise made it happen. But our new platform and agent helped enable it.

Sometimes, changing the phase means sacrificing momentum now to build leverage for later. That takes guts. But it’s part of the process.

Lifters Adjust, Too

Smart lifters know when to change gears. That might mean trying a new program — like shifting from 5/3/1 to Juggernaut. Or letting an AI-based plan like JuggernautAI retune based on fatigue and bar speed. Sometimes it means getting a personal coach, or joining a lifter forum to troubleshoot sticking points.

What matters is sticking to the discipline. You’re still logging reps. Still progressing with intention. The ritual stays — the phase shifts.

Set a Goal to Stay Moving

During COVID, lifting alone in my garage, motivation came from an unexpected place: signing up for a powerlifting meet. I wasn’t planning to set records. But having a date on the calendar gave me focus.

That meet — and the ones that followed — taught me something else: competition isn't about beating others. It's about beating the weight. And the community is there to help you do just that. Even your "opponents" cheer you on.

The same lesson applies in product.

When you're feeling stuck, *talk to your users. Talk to your peers.* They'll remind you why you're doing it. And they'll help you find the next rep.

Even when you change the phase — the load, the format, the plan — your mission stays. Your ritual stays. The growth continues.

Growth That Holds

There's a difference between strength that shows up... and strength that holds up.

That difference comes from experience. From failure. From learning the hard way.

We saw it earlier in the story of *Hafthor Björnsson* — the world's strongest man who tore his pec pushing beyond safe limits. He later called it his "biggest mistake" and rebuilt with a smarter plan. The same pattern shows up

again and again in high-level lifting. Even *Stefi Cohen*, one of the most decorated lifters of her generation, has shifted focus toward long-term health, adjusting how she trains, eats, and recovers — not out of fear, but out of wisdom.

These lessons apply everywhere high performance matters.

Product Example: Build Fast, Learn Faster

At **Endgame**, we learned a critical lesson well before the world-shaking CrowdStrike outage of 2024. We had shipped a powerful capability: the ability to update our machine learning-based malware prevention engine out of cycle from the main product release. It meant our users got faster protection. They didn't have to wait for a full product upgrade.

It was the right feature — but it had an unintended consequence.

During Black Friday at one of our retail customers, an update disrupted point-of-sale systems. Nothing catastrophic. But enough to make an impact.

We fixed it fast. We added guardrails. We changed our rollout process. And we never made the same mistake again. More importantly, we kept the feature — and

improved how it worked *for the user*, at their scale, in their implementation.

We learned that the best product capabilities don't just "work." They hold up.

They account for real-world usage, edge cases, and the 80/20 rule — that 80% of issues will come from 20% of scenarios. And we built resilience into the system because of it.

The Product Takeaway

Your best ideas may cause friction if they're not paired with empathy.

It's not enough to be right. You need to be reliable.

And the only way to get there is through **reps that test the system**, expose the edges, and give you time to learn and evolve. This is how mature products — and mature lifters — endure.

6

The Mission Demands Recovery

The Biology of Bounce Back

Recovery isn't retreat. It's reinforcement. Whether you're lifting heavy or leading strong, progress isn't just about what you do. It's about what you recover from. And yet, in both gyms and boardrooms, recovery is often misunderstood as weakness, wasted time, or something you'll "get to later." When stress hits or deadlines loom, recovery is usually the first thing cut.

That mindset breaks things.

This section makes a different case. It's an often-repeated phrase, "it's a marathon, not a sprint," but even marathons have aid stations. Through real-world examples—from elite athletes to burned-out teams, from space shuttles to software cycles—we'll see how recovery isn't the opposite of performance. It's what makes performance possible.

You'll notice a pattern: the strongest systems, bodies, and teams are built on rhythms, not sprints. And the ones that fall apart? They usually forgot to rest.

The Years Recovery Vanished

During the pandemic, the commute disappeared. And with it, the last buffer between work and rest. Working from home seemed like a win, until days began with a Slack notification and ended with Zoom still open. Lunch breaks were replaced by back-to-back meetings. Evenings bled into late-night emails. There was no off switch, just a lower screen brightness.

People didn't stop working. They stopped recovering.

It wasn't the difficulty of the tasks. It was the absence of boundaries. Parents toggled between spreadsheets and homework help without leaving their chairs. Managers tried to hold their teams together through webcams and chat threads. Even rest became another thing to optimize—a new podcast, a sleep tracker, a mindfulness app squeezed in before another call.

Burnout crept in sideways. Not as a dramatic collapse, but as a slow erosion of clarity, patience, and presence. Work got done. But purpose got blurry. People didn't crash. They faded. Not because they cared too little, but because they were never given space to come up for air.

Recovery wasn't just missing. It had been overwritten.

Mat Fraser and the Discipline of Recovery

That same lesson echoed far beyond the home office.

Mat Fraser, five-time CrossFit Games champion, wasn't always the strongest on paper. But he outlasted everyone. His secret? Obsessive recovery. Fraser treated recovery like training itself—tracking sleep with Whoop data, measuring heart rate variability daily, managing his nutrition down to the ounce. His training sessions were brutal, but his rest protocols were just as structured. Massage, mobility, ice baths, light movement on off days —nothing was left to chance. He didn't see rest as weakness. He saw it as the difference between surviving and dominating. In his own words:

*"I do everything right, especially the **boring** stuff."*

That "boring stuff" is what let him stay healthy, adapt faster, and build a career that eclipsed stronger but less disciplined competitors.

Basecamp's Built-In Cooldown

The same mindset showed up in software development. Basecamp built recovery into their operating rhythm. After every six-week product cycle, where teams

focused on delivering something real, they took two weeks off from planned work. These cooldowns weren't vacations, but a time to reset, fix bugs, explore ideas, or recover team energy. There were no deadlines, no pressure to deliver. This wasn't laziness; it was strategy. Founders Jason Fried and David Heinemeier Hansson believed sustainable work required ebb and flow. Without that rhythm, quality would slip, and burnout would rise. Instead of asking their team to be superhuman, they asked them to be consistent and gave them the space to do it.

G2 Esports and the Power of Tournament Recovery

The power of recovery isn't just for traditional sports or software teams. In the high-stakes world of esports, teams that build recovery into their cycle gain a lasting edge.

G2 Esports, one of the most successful organizations in Valorant, is a prime example. After a disappointing showing in 2023, the team deliberately adjusted its approach — incorporating more deliberate cooldown time between matches and reducing the number of high-pressure scrims. The goal wasn't to train less. It was to train smarter. To recover. To breathe.

The results spoke for themselves. G2 won the VCT Americas Stage 1 title in 2024, emerging as the top team in their region. They followed that with a deep international run at *Masters Bangkok*, one of Valorant's premier global tournaments where the best teams from every region compete for international dominance. G2 reached the Grand Final, pushing T1 to a full five-map series. While they ultimately fell short of the trophy, their ability to perform at the highest level, tournament after tournament, was proof of a system built for endurance.

The Challenger Disaster and the Tragic Cost of Skipping the Pause

And when recovery is missing, the cost isn't just performance. It could mean peoples' lives.

As we wrote in *Rituals Over Rules*: "...our goal isn't to meet an investor's timeline or an executive's forecast. It's to build the best product to solve our user's mission." That perspective doesn't just apply to shipping code. It applies to every system that carries real risk.

In 1986, engineers at Morton Thiokol warned... that cold weather could compromise the space shuttle Challenger's O-ring seals. The data was there. The risk was known. But leadership, under pressure to meet a schedule, pushed forward anyway. The launch went ahead, and seventy-three seconds later, the Challenger

exploded, killing all seven crew members. The physical cause was the O-ring failure. But the systemic cause was the inability to pause. There was no cultural space for dissent, no moment to reflect. The organizational systems were sprinting—politically, reputationally, and economically—with no built-in mechanism for recovery or review. The result wasn't just a tragedy. It was a preventable failure brought on by the absence of pause.

Evernote's Drift Without Reflection

That same cost plays out in the digital world. Evernote was once the darling of productivity tools, with a loyal user base and a clear mission. But as it scaled, it kept adding: more features, more integrations, more complexity. There was no time to reflect, consolidate, or revisit its core value. Internal teams were stretched thin. Technical debt piled up. Bugs increased. The product lost its identity under the weight of unchecked ambition. And as the team chased innovation without rest, users quietly left for simpler, clearer alternatives. Evernote didn't fail overnight. It drifted, feature by feature, update by update, away from what had made it valuable in the first place. All because recovery was never part of the plan.

Protect the Pause

Recovery doesn't happen by accident. It has to be built in, programmed into cycles, protected by boundaries, and respected as essential. The weight you can lift doesn't just depend on what you trained. It depends on whether you recovered.

Strength isn't built in the moment you push. It's built in the space between pushes, if you protect it.

Recovery is strategy. It's not the opposite of work — it's how the work keeps getting done. Whether you're building muscle or building a product, the principle holds: growth requires rest. And not just any rest — intentional, protected, repeated. When you plan for recovery, you're not falling behind. You're investing in your future resilience.

Because the mission doesn't just demand effort. It demands endurance.

Boundaries Are Barriers to Burnout

Boundaries aren't barriers. They're rails. Not to box us in, but to keep us from going off track.

In lifting, the rails are obvious: rest days, deload weeks, technique sessions. You can grind all you want, but if you

don't recover inside the structure, the system breaks. Not always with a snap. Sometimes just with a slow slide into fatigue you can't name.

In product, the rails are harder to see. And harder to hold.

This work is rarely clean. It shifts constantly between strategy and triage: vision decks one moment, sales escalations the next. Context-switching becomes the default. What you planned often gets pushed aside by what the business demands now.

Instead of pretending we can block out perfect “no meeting days,” the better move is to create rails that expect the chaos but still protect time to think clearly.

Call them strategy blocks. Step-back time. Mental deloads. The name doesn’t matter. The intent does: to build in space that lifts your head above the noise and lets you make intentional decisions, not reactive ones.

Because rails aren’t rigidity. They’re protection. For focus. For energy. For the mission.

Netflix and the Space to Reflect

Even in a high-performance culture like Netflix, boundaries around strategic thinking are built into the

system. Product leaders are expected to regularly pause and reflect — not just on what they’re building, but on who’s building it. Their well-known “Keeper Test” asks a simple but revealing question: Would I rehire this person for this role today? Answering that honestly takes courage. But it also takes time. Time to zoom out. Time to get quiet enough to think clearly. In a culture that thrives on speed, those reflections are what keep the team aligned, accountable, and evolving.

Step-back time doesn’t have to be silent. It just has to be sacred.

Sustainable Strength

Recovery is not a reward. It’s part of the plan.

In strength training, the most disciplined athletes don’t just train hard — they track what fuels the work. Calories. Sleep. Macros. Because performance isn’t just about what you do in the gym. It’s about what you support outside of it. That’s why tools like MacroFactor have become essential for serious lifters: not for vanity, but for visibility. To make sure every training block is fed with what it needs to grow.

In product, the same principle applies. You can’t build a resilient team without knowing what you’re feeding it. That’s where tools like Jira, Aha!, or Linear come in — not missionbuilt.io

just as ticket trackers, but as visibility engines. They let you zoom out. Spot overtraining. Identify where strategic work is undernourished. Recovery in product isn't just about taking breaks. It's about sustaining the energy to build well over time.

Because burnout doesn't usually come from sprinting. It comes from sprinting without a cycle.

That's why elite lifters use periodization — planning not just the next workout, but the next month, quarter, and year. They alternate intensities. Deload. Peak. Rebuild. Not because they're lazy. Because they're playing the long game.

Sustainable product teams follow the same rhythm. They don't live in permanent crunch. They push when necessary, then deliberately create time to reflect, retool, or recharge. They respect the difference between urgency and importance. And they recognize that more activity doesn't always mean more progress.

When you track your macros, you don't just know what you're eating. You know what you're aiming for.

When you track your product macros — the balance of innovation, maintenance, user research, and downtime — you build not just velocity, but viability.

One example of this kind of tracking in action comes from Atlassian's *Team Health Monitors*. These are not daily scrums or sprint reviews. They are recurring, focused check-ins designed to assess the long-term health of a team. Each one reviews key areas: clarity of roles, alignment on goals, confidence in decisions, pace of delivery, and engagement with stakeholders. Teams score themselves on a red-yellow-green scale to visualize patterns over time. The point isn't to be perfect. It's to stay aware — to notice stress points early, to invest in what's working, and to surface risks before they derail momentum. It's macro tracking for the makers. Less about this week's burndown, more about whether the system is still healthy enough to support what's next.

Because the strongest teams aren't the ones who go the hardest.

They're the ones who stay strong the longest.

Recovery Is a Leadership Skill

If you want to go the distance — as an athlete, a builder, or a leader — you can't just train intensity. You have to train sustainability.

Recovery isn't about stepping back. It's about stepping up — with clearer focus, steadier energy, and decisions made from purpose instead of panic.

In the gym, you build this with sleep, nutrition, deloads, and reflection.

In product, you build it with boundaries, cooldowns, strategy days, and macro awareness.

You build it when you protect the pause. When you track more than just tickets. When you care not just about the sprint, but about who's still standing after it.

Because the mission won't slow down. But you can build systems — and teams — that last.

Aside: Recovery Creates Room for Perspective

At this point in the book—halfway through, and at a chapter all about recovery—it's worth pausing to reflect on something that too often gets misunderstood or misused in product work: DEI.

The acronym itself has been politicized and weaponized. But the principle behind it? It's not just relevant to building good teams—it's essential to building great products. Because the enemy of clarity isn't chaos. It's assumption. And when your product is only shaped by people who already know how to use it, you stop seeing what's broken.

Think about the average enterprise tool. Bloated. Confusing. Clunky. Not because the teams behind them don't care—but because they've learned to work around the flaws. They stop noticing friction because they've internalized the shortcuts. And then they add more complexity without questioning the base layer. That's not innovation. That's calcification.

Remember Apple's infamous iPhone 4 moment? When reception dropped if the device was held a certain way, the official response was: "You're holding it wrong." That wasn't just a PR misstep. It was a failure of empathy. A product built with a single use case in mind—and a team unwilling to question it.

This is where diversity becomes a design principle. It's not about optics. It's about inputs. When you bring in voices that don't match your defaults—different backgrounds, different abilities, different ways of thinking—you create friction that reveals blind spots. You build products that more people can actually use.

The Medici Group teaches a powerful example of this in their innovation workshops: hospitals learning from Formula 1 pit crews. Two entirely different fields—surgery and motorsport—sharing best practices to reduce patient handoff time in operating rooms. The result? Faster triage. Fewer errors. Lives saved. That's

what happens when you step outside your bubble and let diversity expand your playbook.

And sometimes, that kind of inclusion is quieter—but just as critical.

Respect the Pauses

Diversity comes in many forms, including language. At Elastic, where teams span the globe, English is the default for most business communication—but it's not the first language for many Elasticians. That matters more than most people realize.

In a leadership training, Madhura Chopda, Senior Director of Engineering, shared what happens cognitively when you're a non-native English speaker in a fast-moving meeting. You listen. You process. You translate. You form a thought. You translate again to respond. That's not delay. That's effort. And if you interrupt the pause where that work happens, you erase the contribution before it can surface.

Respecting the pause allows full participation. It gives space not just for language translation, but for different processing styles, communication rhythms, neurodiversity, and accessibility needs.

Internally, we called it the curb-cut effect. Curb ramps were designed for wheelchairs. But now they help everyone—people with strollers, luggage, bikes, injuries. What was built for access became a better default for all.

Respecting the pause works the same way. It's a small behavior that opens the door to a much wider room.

7

Train the Engine, Not Just the Output

Work Capacity Over Window Dressing

There's a difference between looking strong and *being* strong. You see it in the gym all the time: the lifter who chases their one-rep max every week, treating PRs like progress reports. Impressive — until the plateaus hit or the pain sets in. What you don't see as often is the lifter dragging a sled after squats. Doing walking lunges in the cold. Rowing in the morning and hitting their accessories at night. That's the engine work. That's capacity.

If you want to go far, build the lungs before the legs.

In the training world, this is called **GPP — General Physical Preparedness**.

It's not just a warmup or conditioning. It's the foundation for everything else.

GPP improves recovery between sets and sessions. It builds joint integrity, tissue resilience, and work tolerance. It makes your system *ready* — ready to train harder, recover faster, and adapt more consistently. And yet, in many strength circles, it's dismissed. You'll hear things like:

- “Combo sets are my cardio.”
- “I don’t have time for fluff — just the main lifts.”
- “If it doesn’t add pounds to the bar, it’s a distraction.”

That’s the specialization trap. And it’s seductive.

Specialized programs promise fast results. They’re targeted. Efficient. And if you’re new to lifting — or building a product — specialization feels like the smart move. Just do *the thing* you want to improve, and do more of it.

Until it stops working.

Until your joints hurt. Your energy fades. Your momentum disappears. And more than that — you burn out.

Because repetition without variation isn’t just physically risky — it’s psychologically draining. The mind, like the body, thrives on challenge and change. Without

variation, even the most disciplined lifter starts dreading the next workout. Even the most motivated developer begins to disengage.

We've seen this in teams that *only* work on performance. Or *only* ship new features. The work becomes mechanical. Morale suffers. Curiosity fades. They're strong in one motion — but brittle under real-world load.

That's why smart teams — and smart lifters — build in variation.

Like the **6–2 cycle** popularized by Basecamp: six weeks of deep, focused product work followed by two weeks of cooldown — space for maintenance, reflection, and system health.

It's not a break. It's capacity training.

A rhythm we explored in **Chapter 6: The Mission Demands Recovery**, in the section titled *Basecamp's Built-In Cooldown* — and one that other companies have since adopted and evolved because it works.

GPP isn't glamorous. But it's what lets you train hard — and show up strong — again tomorrow.

That's what Alex Viada understood. A biomedical engineer and hybrid athlete, Viada rejected the false binary between strength and endurance. He trained for both — and thrived. While most lifters feared that conditioning would eat their gains, Viada built an aerobic

base robust enough to support heavy squats *and* sub-5:00 miles. His results weren't just impressive — they were repeatable. Durable.

You see the same principle in product teams. Linear didn't try to dazzle with a full feature suite out of the gate. Instead, they trained their engine: performance. Every click, every transition, every load state was built to feel instant — even offline. That invisible investment became their advantage. Users stayed focused, fast, and frictionless. And while competitors stacked features, Linear scaled loyalty.

But the obsession with output can take an even darker turn — in the gym *and* in the market.

Some lifters now chase not even performance, but the appearance of performance.

Injections like **synthol** or site-enhancing oils artificially inflate muscle volume — biceps, shoulders, traps — to create the illusion of strength. No increased output. No increase in health. Just ballooned tissue under the skin. These are vanity muscles, pure and simple. And in too many cases, they don't just harm the lifter's reputation — they destroy their body.

Heart failure. Sepsis. Amputation. Even death.

When looking strong matters more than being strong, the mission is already lost.

And in product? It's not so different.

Teams inject vanity metrics into their roadmaps: flashy dashboards no one uses. AI features for the pitch deck. Slideware that outpaces software. It might look good in the moment — but behind the scenes, the engine's breaking down.

You can't fake work capacity. You can fake a feature. You can fake a max lift. But you can't fake the foundation.

Burnout is what happens when you try to specialize your way out of exhaustion.

Capacity is what happens when you train for the long haul.

The ones who last don't just train the output.
They train the system behind the output.

Internal Strength for External Load

Strength isn't just about what you can lift. It's about what you can *withstand*.

Anyone can show up strong once. But the mission demands more than that. It demands repeatability. Resilience. The ability to show up strong again and again, under pressure, under uncertainty, under load.

That kind of strength doesn't come from testing yourself every day. It comes from building yourself for the test — from the inside out.

We know this intuitively in lifting. If you neglect your stabilizers, your joints, your breath control, your bracing — it doesn't matter what your numbers look like. You'll break. The biggest total in the gym means nothing if your back gives out the next week. True strength isn't just output. It's the integrity of the system supporting it.

The best lifters don't chase the max. They train the margin.

That's what Amanda Lawrence understands. A multi-time world champion powerlifter, Amanda is famous not just for her records — squatting over 600 pounds raw at 84kg — but for her poise under pressure. She lifts with precision, with control, and with composure. And that's not an accident.

Amanda has spoken about how she visualizes every meet lift weeks in advance. She rehearses the rituals. She manages recovery like a professional sport. She doesn't just train her muscles. She trains her nervous system, her mental process, her ability to stay composed under load.

That internal strength — not just quad size or grip — is what lets her hit massive lifts on the world stage, even when the bar bends and the cameras are rolling.

They trained the tendons of the product, not just the muscle.

The same applies in product.

Notion wasn't always a darling. In 2015, they were nearly dead. No traction. Technical debt piling up. The team had two choices: patch the pain and push forward, or stop, go dark, and rebuild the core from scratch.

They chose the hard route. Two engineers rewrote the entire editor engine over 18 months. No releases. No hype. Just deep technical investment in a stronger, more flexible system. They bet that the internal strength of the product — not surface-level features — would determine their long-term viability.

They were right.

That rebuilt core became the foundation for everything that followed: blocks, embeds, team permissions, scalable collaboration. The invisible architecture made possible everything the user eventually saw. What saved Notion wasn't a sprint. It was a rebuild.

Throughout my career, I've seen how scale and stability work is often dismissed as unexciting — not flashy enough for slides, not "pressable" enough for headlines. Some leaders fall into the trap of always needing something new to talk about. Something shippable. Something that demos well.

But that pressure leads to bloated products, incoherent strategies, and worst of all — users who suffer when the product finally gets real-world load.

When everything matters most, the engine breaks.

Here's the truth: capability refinement is *not* the boring work.

It's the hard work.

It's where the most difficult engineering problems live.

It's where the most intricate, high-stakes user requirements get applied — not in the pitch deck, but in the heat of usage.

Building a reliable core doesn't mean every startup needs to design for a Fortune 10 on day one. But it *does* mean they need to carve out time in the roadmap to move deliberately toward that direction. They need to ask the hard questions early — not just "What are we building?" but "Who are we building it for?"

And just as critically: "Who are we *not* building it for?"

You cannot build for every user in every vertical in every geo all at once.

Trying to do so leads to brittle systems and vague products that serve no one well.

As we said in earlier chapters: **laser focus is strength**.

Define your product requirements. Define your user.

Define the scale. Define your edge cases and your tests.

Then *do the work* — especially the work that doesn't make the press release.

Because that's where real durability comes from.

Not the big reveal.

The boring reps.

I've lived a version of that story myself.

Early at Endgame, after CEO Nate Fick made the difficult but necessary decision to narrow our product portfolio to one core offering, engineering leader Tony Meehan and I found ourselves at a crossroads.

The product we had was powerful. It delivered real results and we had major enterprise deals on the horizon. But the architecture wasn't built for scale. The team had shown clearly that it wouldn't hold as we moved into larger opportunities.

We could have coasted. Closed deals. Deferred the hard part.

But we knew that if we built a base of users on that foundation, the eventual transition would be expensive and painful.

So in early 2016, we made the call to rebuild. We moved the frontend to React, implemented a new queueing system with NATS.io, and redesigned how the endpoint agent communicated. All of this had to be done in under six months — alongside new features — for a critical delivery with the U.S. Department of Defense.

It was brutal. Long hours. Hard conversations. Constant doubt.

But participating in that live DoD exercise, watching operators succeed with what we had built, it was clear. The effort had been worth it.

The new architecture held. It scaled. It supported the mission until we joined Elastic in 2019.

| Internal strength isn't flashy. But it's what gets you through the weight that would break someone else.

What Powers You?

The fastest engines in the world still go nowhere without fuel.

You can have world-class systems, perfect technique, and an unbreakable schedule. But without something driving you — a reason to push through when it's hard, a mission that makes the effort matter — the engine stalls. Strength without fuel is just potential. It goes nowhere.

This is the third layer of performance.

After physical preparation and mental resilience comes **purpose**.

The *why* behind the work. The force that powers the engine when the load gets heavy and the results are still far away.

You can see this difference on the platform.

Some lifters are chasing numbers. Others are lifting for something deeper — to test themselves, to prove something to a younger version of who they were, or to inspire someone watching from the crowd. The same weight moves differently depending on what's driving it.

The same is true in product work.

Teams that ship the fastest aren't just well-organized. They're aligned around a purpose they care about. They understand the mission. They know what their work *means* to the people who use it. And that makes them more resilient, not just more productive.

I When the fuel runs low, purpose drives on.

Take **Kristy Hawkins**.

Before she was a record-breaking powerlifter, she was a biochemical engineer at Genentech. Her story isn't about switching careers — it's about doubling down on curiosity and mastery. Whether in a lab or under a barbell, Kristy trains with the same energy source: a relentless drive to understand, improve, and own her progress.

She's not powered by comparison. She's powered by process.

And that's what makes her strength sustainable. You don't win in two domains over decades by accident. You win by building an engine that *wants* the work.

The same goes for teams.

Look at how **Superhuman** built their product.

Founder Rahul Vohra didn't just want to make a faster inbox. He wanted to build *the* fastest, most delightful email experience ever created. That goal wasn't vague or

corporate. It was clear, obsessive, and emotionally charged.

They didn't launch until users said they'd be "very disappointed" if they lost the product.

That wasn't marketing polish. It was purpose, measured.

And it gave the team an engine.

Every decision — from UI transitions to onboarding flow — was shaped by that internal fire: delight. Clarity. Joy. The result wasn't just speed. It was focus. The product knew what it was, and the team knew why it mattered.

Capacity is mechanical. Mental strength is adaptive. But purpose is emotional. It's fuel.

That's why it matters *what* you're building strength for.

You can build a massive work capacity and still burn out if you're not connected to what it's in service of. You can become mentally tough and still drift if the mission isn't clear.

Even strong people lose their way when they forget why they're moving in the first place.

The question isn't just: *Can you go further?*
It's: *Why would you?* That's the check-in.

Ask any seasoned lifter what keeps them going, and the answers get quieter over time.

It's not the meet. Not the total.

It's that the ritual grounds them.

It makes them better.

It reminds them of who they are.

Ask a product team what keeps them shipping when everything breaks, and if they're still smiling, it's because the mission means something to them. They've felt the user's pain. They've seen the real-world impact. They've tasted the outcome of building something that matters.

So, what powers you?

Not just when things are working. Not just when people are watching. But when the bar is stuck. When the sprint is a slog. When the road is long and there's no applause in sight.

That's the kind of test SERE training is designed for. In the U.S. military's Survival, Evasion, Resistance, and Escape course, individuals are pushed to their psychological limits — deprived of sleep, isolated, interrogated. It's not just a test of skill. It's a test of *resolve*. Of remembering the mission when the environment is engineered to make you forget.

The ones who endure don't have better gear or more talent.

They have clarity.

They know what they're fighting for.

That's your fuel.

That's your engine.

That's the difference between showing up when it's **easy**, and showing up when it **counts**.

8

Decisions Are Made Under Load

Stress Tests the System

Stress doesn't break the system. It exposes what was already weak.

We often talk about “testing under load” as a strength-building concept. In the gym, it’s simple—your ability to move weight when you’re tired is the real signal of progress. But in product leadership, systems design, and incident response, stress testing is often overlooked until it’s too late.

The **Log4j vulnerability** in December 2021 made that brutally clear. A zero-day exploit in a widely used Java library allowed remote code execution with minimal effort. Suddenly, nearly every organization had the same urgent questions:

- Are we exposed?
- Where is Log4j deployed in our environment?
- Do we have visibility into historical access or exploitation attempts?

This wasn't just a code issue. It was a test of organizational readiness—technical, operational, and strategic. Teams that had practiced incident response, mapped their architecture, and built cross-functional trust could move with purpose. Others stumbled, not because they didn't care, but because they'd never trained for the moment.

It was the backup problem, writ large. Many companies back up data religiously, but few test whether they can *actually recover it*. Even fewer know how long recovery will take, or what their **RTO (Recovery Time Objective)** or **RPO (Recovery Point Objective)** really are. And as **ransomware** surged in recent years, those blind spots turned into existential threats. The lesson was hard-earned: backups are only as valuable as your ability to restore *fast*.

Log4j forced a parallel reckoning. In many cases, malicious scanning began months before the public disclosure but most organizations didn't retain accessible data that far back. They had cold storage, sure—frozen or tiered off for cost savings—but had

never practiced bringing it back online quickly for investigation.

When the pressure hit, teams had to figure out—often in real time—how to thaw data, rehydrate logs, and search for infection markers across old records.

Some had tested this process before. They knew the commands, the timing, the quirks of their tooling. They'd trained like operators. Others... hadn't. Some couldn't search more than 30 days back. Others found that recovering historical logs was so cumbersome it may as well not have existed. Those companies weren't just missing data. They were missing a **disaster recovery plan** for threat analytics.

At Elastic, customers who had previously rehearsed these data flows—archiving, restoring, and hunting—were able to quickly triage and verify impact, drawing from months-old telemetry. They weren't just ingesting logs. They were building institutional memory.

And this mirrors what elite military units do when they train decision-making under fatigue. In Navy SEAL training or Army Special Forces SERE programs, candidates are made to perform operational tasks after sleep deprivation, cold water exposure, and physical exhaustion.

Not to break them—but to reveal what's already broken.

In those moments, you don't rise to the level of the manual. You fall to the level of your defaults. And the only way to improve those defaults is to train them, deliberately, under controlled stress.

It's no different in product and platform work. You might have a dashboard for monitoring, a workflow for restores, a playbook for incident response—but unless you've *tested them when it matters*, they're just ideas. Stress finds the truth.

So whether you're squatting on trembling legs after five working sets or trying to rehydrate petabytes of logs to confirm an intrusion:

The system only works if it works under pressure.

This isn't a new idea—it's one we introduced back in **Chapter 2: Built Through Reps**, where we explored the quiet, compounding power of systems over goals. There, we referenced **James Clear**, who wrote in *Atomic Habits*:

"You do not rise to the level of your goals. You fall to the level of your systems."

That quote framed our conversation around sustainable progress—how strength is built not by heroic effort, but by consistent execution of a well-designed system. Here, under the weight of stress and urgency, the meaning deepens.

Goals vanish in crisis. Roadmaps collapse under pressure.

What remains is the infrastructure you've trained, tested, and trusted.

Whether that's a disaster recovery pipeline, a muscle memory forged in fatigue, or a culture that knows how to move without waiting for permission—**it's your system that carries you.**

Resilience isn't reactive. It's built.

Deliberately. Daily. And under load.

Clarity Beats Certainty

You don't need to know the outcome to know how to act.

There's a myth in leadership—and in lifting—that success comes from always having the right answer. That if you just gather enough data, wait long enough, or plan hard enough, certainty will arrive and carry you through.

But the best teams, the best lifters, the best products—they don't run on certainty. They run on clarity.

You see it clearly in moments of disruption. In early 2020, when the COVID-19 pandemic rewrote every assumption about business, teams that waited for complete information froze. But the ones who moved quickly didn't do it recklessly—they did it with intention.

- **Zoom** opened up free access to educators around the world.
 - So what? Within weeks, daily meeting participants skyrocketed from **10 million in December 2019** to **300 million by April 2020**, securing Zoom's role as the go-to platform for remote communication.
- **Peloton** overhauled its entire supply chain when demand outpaced capacity.
 - So what? By Q4 of 2020, Peloton reported **232% year-over-year revenue growth** and added **over 1 million new subscribers**, despite early shipping delays—because it chose to act early.
- **Notion** refocused messaging to meet the remote work moment head-on.
 - So what? In 2020, Notion's user base grew from **1 million to over 4 million users**, aided by a pivot to freemium and a surge in international adoption driven by product-led, mission-aligned messaging.

None of them knew how long the shift would last. They couldn't predict the full impact. But they didn't need

certainty—they had clarity of mission. That clarity became a compass.

This is the same clarity that high-level lifters use when training with RPE—Rate of Perceived Exertion. Instead of sticking to a fixed number on the bar, they assess how each set feels in real time. On some days, that means pulling back to preserve recovery. On others, it means pushing heavier than planned.

It's not guesswork—it's informed adaptation.

It's clarity in action, even when conditions change.

We talked about this earlier in the book when introducing the **OODA Loop**—*Observe, Orient, Decide, Act*. Originally developed by military strategist John Boyd to train fighter pilots for split-second decisions in combat, the OODA loop isn't about perfection. It's about momentum. You act, assess, and adjust faster than the environment can overwhelm you.

This model doesn't just apply to military operations. It applies to product releases, user feedback, platform incidents, and even organizational change. You're not waiting for every signal to align—you're moving with awareness, grounded in your principles, and fast enough to respond before inertia sets in.

And this is where clarity beats certainty.

- Certainty waits.
- Clarity prepares.
- Certainty asks, “Are we sure?”
- Clarity says, “Here’s what we do next.”

You don’t always need a perfect map if you’ve trained your compass.

Let’s be honest: **Agile** is a loaded word.

Some teams swear by it. Others roll their eyes. And more often than not, the problem isn’t with the philosophy—it’s with the implementation.

Too many teams have had Agile forced on them by someone who read a textbook but never wrote a line of code. The rituals become rigid. The flexibility becomes performative. And the spirit gets lost in the ceremony.

I’m a firm believer in the **Agile mindset**—but not the dogmatic, by-the-book version. True agility means responding quickly to new information. It means iteration with intent. But that only works if the team has clarity. Without it, Agile devolves into a flurry of half-finished tickets, context switches, and directionless pivots.

The point was never to abandon focus.

The point was to stay mission-aligned, laser-focused on the “why,” while remaining flexible on the “how.”

Where I most often see Agile fail is when it becomes a cover for **indecision**. When teams confuse agility with ambiguity. When the lack of a clear goal gets labeled as “keeping our options open.”

I also see it fail when people demand certainty before committing to action. Even today—with more data than ever before—certainty is rare.

We have telemetry, user interviews, funnel analysis, Salesforce records, analyst feedback...

But the decision on what to build next is still more art than science.

| If there was a playbook to follow, we'd be project managers. But we're product managers.

Our craft isn't in perfect foresight. It's in translating user pain into progress. It's in surfacing the “why” beneath the “what.” It's in seeing the pattern behind the request.

And as **Nathaniel Fick** wrote in *One Bullet Away*—his memoir about the transformation from Ivy League student to Marine officer leading combat missions in Afghanistan and Iraq:

| “Complex ideas must be made simple, or they'll remain ideas and never be put into action.”

That's our job. To take everything we know—the signals, the inputs, the intuition—and turn it into action. Not after a perfect analysis. Not when every detail is known. But when it's time to move.

We do that not by being certain—but by being clear.

Clear on the mission.

Clear on the user.

Clear on what matters most right now.

Because clarity moves.

Certainty stalls.

And in the fog of delivery, feedback, and shifting priorities, the job is not to predict perfectly. It's to navigate **intentionally**.

Training for Chaos

We don't rise to the level of our plans—we fall to the level of our preparation.

In the last chapter, we explored the idea of **training the engine**—building general physical and mental capacity, not just specific skills. But that alone isn't enough. What happens when the plan vanishes? When you're under pressure, short on time, and forced to act without hesitation?

That's when preparation has to become instinct.

I learned this firsthand as a **U.S. Army paratrooper**.

Before you ever board a plane, you train on the ground for weeks. You drill every movement until it's embedded—how to hook up your static line, how to check your gear, how to exit, how to land. You jump from towers. You simulate the aircraft. And you practice failure—not because you want it, but because it's coming.

You want to fail when you're four feet off the ground—not 800.

We don't take our first flight until the basics are second nature. Because stepping out the side door of a C-130 at 800 feet, in pitch black, isn't natural. It's against every self-preservation instinct. And in that moment, you don't want to be thinking through a checklist. You want to be moving.

If your main chute doesn't deploy, you have **9 to 10 seconds** to respond before impact. That's it. There's no time to think. You just act.

I still remember my first jump. The roar of the engines. The rush of the open door. And when we stood up, hooked up, and waited for the green light—I didn't feel fear. I didn't feel stress.

I felt clarity.

I knew what to do. And I executed. Because we had trained for it.

That same mindset exists in the best teams—even in tech.

At Netflix, **Chaos Monkey** was built to deliberately shut down random services in production—to ensure that both systems and people could adapt under pressure. They didn't just hope their systems were resilient. They trained them to be. And the results were measurable:

*Netflix prevented **80% of potential outages** through the learnings surfaced by Chaos Monkey and the broader chaos engineering discipline. These weren't hypothetical flaws—they were real vulnerabilities, caught early because someone had the courage to test failure on purpose.*

This wasn't an act of sabotage. It was an act of preparation.

You don't find out if you're ready in the middle of a real incident. You find out beforehand—if you're willing to simulate the storm.

You see this in sport, too. During the **CrossFit Open**, athletes receive surprise workouts announced just days

before competition. Movement patterns shift. Equipment combinations vary. Sometimes it's a short, explosive sprint. Other times, it's a long, grinding engine test. You don't know what's coming—and that's the point.

This unpredictability isn't accidental. It reflects one of CrossFit's founding principles: *prepare for the unknown and the unknowable*. Because in real life—just like in combat, product development, or incident response—you don't get a playbook with advance notice. You're asked to perform under pressure, with incomplete information and limited control.

By designing chaos into the format—new standards, awkward movement pairings, strange pacing demands—CrossFit forces athletes to confront not just their physical readiness, but their emotional and mental adaptability. The top performers aren't just strong. They don't just move well.

I *They stay composed when the plan disappears.*

And that's the test. That's the transferability. The Open doesn't just reward fitness. It rewards clarity under fatigue, pacing under uncertainty, and strategy when the rules shift.

The chaos isn't a bug—it's the feature. It reveals what structure alone can't.

This belongs in the same category as **paratrooper training** or **chaos engineering**. You don't need bullets or breakage for chaos to be real. All you need is volatility, pressure, and a demand for action.

And when that moment comes, the best don't panic.

They breathe. They move. They decide.

I want to pause here and share something personal.

I've never claimed to have done the hardest thing in the military. Many others have faced more direct danger, made life-or-death decisions in the field, and carried far heavier loads. Nate Fick, whose story we referenced earlier, is one of many. But I've also learned that **you can't live your life comparing hardships**. Chaos shows up in many forms. And every one of us has moments when our training is tested by stress.

For me, that came during my deployments in support of Operation Enduring Freedom and Operation Iraqi Freedom, where I served as an intelligence analyst at the Combined Air Operations Center. My job was to represent U.S. Army interests in daily joint targeting and mission planning—to advocate for the air support our soldiers needed downrange.

That meant walking into rooms where I was often the lowest-ranking person present, trying to secure limited airframes for missions that couldn't afford to fail. There were more missions than aircraft. Everyone in the room wanted to help—but everyone had missions to support. The real pressure wasn't the data. It was the conversation. The negotiation. The weight of knowing that if I couldn't make our case clearly, my brothers in arms might not get the support they needed.

That stress taught me the true value of clarity. Not just in numbers, but in stories. It wasn't the PowerPoint or the packet that made the difference—it was being able to explain why this mattered, what the risk really was, and who would be impacted if we didn't act. I learned young, and under pressure, how to speak across rank and service. How to translate intelligence into insight. And most importantly, how to **give a shit** - loudly, clearly, and with purpose.

One of the awards I received—the Army Commendation Medal—noted that I served in multiple roles normally held by ranks far above mine, including as the BCD Intelligence NCOIC (a Sergeant First Class role) and as an Intelligence Plans Officer, a position typically held by a Major. I don't say that to boast. I say it to underscore this truth:

You don't perform under stress unless you care deeply about the mission and the people it impacts.

That lesson stayed with me. In uniform. In product. In leadership.

It's not process that drives you when the pressure hits. It's **principle**. It's **preparation**. And it's the ability to act with conviction, because you know exactly what's at stake.

You don't have to be on the front lines to know what chaos feels like. You just have to care enough to move when it matters most.

9

Ship It Like You Show Up

Integrity in Small Things

How you do the small things is how you do everything.

Greatness doesn't usually show up where we expect it. It isn't built in the spotlight. Not in PR lifts or flashy product launches. Not in press releases or headlines. Most of what truly earns trust and builds strength happens quietly. In the warmups. In the follow-through. In the care no one notices but you.

From the outside, powerlifting looks simple. Three lifts: squat, bench, and deadlift. Movements most gym-goers already know. That's part of its charm. Anyone can walk into a meet and give it a shot. No gatekeeping. No secrets.

But the difference between showing up and winning isn't about size or intensity. It's about discipline. The small

things. The lifters who win are the ones who move with precision.

The bar comes out of the rack the same way every time. The breath is controlled. The bar path is tight and deliberate. The brace is locked in. The foot pressure is balanced. And they do it the same way whether it's 135 pounds or a max-effort lift.

That's where mastery comes from: consistency under load.

Watch Olympic lifter Lu Xiaojun and you'll see it. People celebrate his explosive strength, but what makes him exceptional is his control. Every rep is intentional. Every rerack is clean. There's no ego in his movement, only precision. He lifts like someone who knows that how you finish matters just as much as how you start.

This is what the best product teams understand too.

Anyone can ship a feature. But not everyone takes the time to check whether the error message makes sense. Not everyone notices if the cursor lands in the right input field. Not everyone asks if the experience feels smooth or rushed.

But when someone does, users feel it. Even if they don't know why. It's the difference between something that works and something that feels right.

At GitHub, engineers fix what they call papercuts - small friction points most people would ignore. They don't wait for permission. They just fix them. And when they do, the product feels better. Not flashier. Just more thoughtful.

At Apple, designers obsess over pixels and motion. Scroll behavior, bounce physics, shadow softness. These are the details most people never consciously notice. But they notice how it feels. And that's the point. The little things aren't just polish. They're the experience.

This isn't fluff. It's care, made visible.

So why doesn't the most full-featured product always win?

Android phones offer more toggles, more options, longer spec sheets. Yet Apple, with fewer features and tighter control, continues to lead in loyalty and user satisfaction.

Think back to Blackberry, when the iPhone first launched. "No one wants to type on glass," they said.

They were wrong. People didn't want keys. They wanted clarity. Simplicity. Something that felt modern and intuitive. Apple gave it to them.

The first iPhone didn't have copy and paste. There was no App Store. No video recording. On paper, it looked incomplete. But in practice, it felt revolutionary. The gestures were smooth. The interface invited you in. It felt alive.

Behind the scenes, Apple's teams spent months tuning details most users would never name. Scroll speed. Button animation timing. The physics of how things moved on screen. Not because it made a better demo, but because it created a better experience.

Other companies added features. Apple focused on how it felt to use.

BMW takes a similar approach. They don't sell the highest horsepower. They sell the experience of driving. The way the wheel feels in your hands. The way the chassis responds to your decisions. That's what makes it premium. It isn't just what it can do. It's how it makes you feel while doing it.

That's the difference.

Great products don't win because they have the longest list of features. They win because they've been shaped by people who care about every detail. People who understand and obsess about the mission.

That mindset shows up outside of product and sport too.

John Cena, before the fame, was a regular at Gold's Gym. He trained hard, but that's not what stuck with people. What they remember is how he re-racked every plate. Wiped down every bench. Left the space better than he found it. No cameras. No applause. Just quiet respect for the work and for the people who would follow.

You don't earn trust with a launch. You earn it with the work no one claps for.

Because when the moment comes, when the lift is on the platform or the product is in the wild, the real question isn't, "Did it work?" The question is, "Does this reflect who we are when no one's watching?"

The Shipping Standard

A clean lift on the platform tells you more than any gym session ever could.

You can move big numbers in training, hit personal records, and feel confident under your own rhythm. But step onto the platform at a meet, and everything changes. The commands are faster. The lights are brighter. The room is louder. The pressure is real. And even with the same weight on the bar, your form starts to slip.

Not because you lack strength, but because you didn't train for that moment.

Building is like training. Shipping is competing.

And competing demands more than power. It requires control, presence, and precision when the world is watching. It also introduces more variables than training ever does. In the gym, you control the pace, the setup, and the rhythm. But on the platform, you might face a cold bar, unfamiliar flooring, a tight timeline, or commands that arrive quicker than expected.

Shipping is no different.

In development, teams test the happy path. In production, users rarely follow it. They skip steps, input bad data, use features in unexpected ways, and still expect everything to work. These edge cases aren't theoretical. They are reality. That's the difference

between building in private and shipping in public. That's what it means to compete.

At Netflix, shipping isn't something saved for the end. It's part of every step. Engineers own their code from the first line to long after it's live. They test it, monitor it, tune it, and fix it when needed. There's no handoff. No deflection. Accountability is shared.

Staging environments mirror production closely, simulating real user behavior and traffic patterns. Netflix teams know you can't expect consistent performance unless you train in the conditions you'll actually face. They don't celebrate heroics after a failure. They celebrate readiness before one. That's how they maintain quality under pressure.

Shipping is a team sport, and the teams that win align around a shared standard of care.

You see that same commitment at NASA.

Before every launch, they hold a Flight Readiness Review. Each subsystem lead, whether in engineering, mission ops, safety, or comms, walks through their area of responsibility and gives a go or no-go decision. If even one person says no-go, the mission halts. No debates. No pressure to push through. Just a clear respect for the standard.

When the risks are that high, trust isn't built on optimism. It's built on preparation, discipline, and the shared courage to pause until it's right.

Even with all that care, sometimes the lift still doesn't land.

In powerlifting, you might feel like you hit a perfect rep, but the judge flashes red. Maybe your depth was just short. Maybe the lockout wasn't fully controlled. Maybe you rushed the pause. The judge doesn't grade your effort. They grade your execution.

You don't get to argue. You adjust.

The best lifters don't spiral when they miss. They listen, learn, and step back on the platform with more precision than before. You miss. You learn. You lift again.

Shipping is no different.

You might release something you believe in, only to hear that it confused users or didn't work as expected. Maybe the problem it solved internally doesn't translate externally. That's not failure. That's feedback. And the only real failure is refusing to respond.

The best teams are not perfect. They are resilient. They listen to signals and respond with care. They don't just

ship once. They ship again and again, with more awareness every time.

We've all seen what happens when something goes out just to meet a deadline. Corners are cut. Quality drops. Bugs slip through. The follow-up work takes longer than doing it right would have in the first place. And users notice.

That isn't shipping. That's recovery.

A real shipping standard doesn't mean moving slowly. It means moving with intention. The checklist exists before the pressure hits. Dry runs happen before the launch. And everyone knows what "ready" really means—not just functional, but finished.

Your standard is your signature.

If you wouldn't sign your name to it, it's not ready.

Lifting the weight is only part of the story. Anyone can move it once. But the lifter who shows up, hits depth, follows commands, and finishes clean is the one who earns respect.

In product, it's the same.

You don't earn trust because you shipped something big.

You earn trust because you shipped something you stand behind.

That's why elite teams don't rush. They prepare. As the U.S. Marine Corps teaches:

"Slow is smooth. Smooth is fast."

It isn't about hesitation—it's about precision under pressure.

You train the way you want to perform. You ship the way you want to be trusted.

Speed without control leads to chaos. But when you move with purpose, readiness compounds into confidence.

That's what creates momentum you can build on, not just scramble after.

Everything Built In

On the surface, powerlifting seems like a simple sport. Three main movements. No elaborate choreography. No specialized gear beyond a belt and chalk. Just a barbell and the strength to move it.

But beneath that simplicity lies extraordinary depth. At the highest level, success doesn't come from brute

force. It comes from total integration - each breath, angle, and cue refined until nothing is wasted.

Lu Xiaojun isn't dominant because he trains harder. He's dominant because his training is complete. His grip, his breath, his core, and his positioning are aligned. There is no wasted motion. Every part of his movement supports the next. He's not just strong. He's unified.

Product teams face a similar challenge. It's easy to chase what's next - the shiny feature, the headline for the release notes. But when something new is added without regard for the whole, it often unsettles more than it improves. Instead of adding value, it reveals what's missing. Instead of delighting users, it creates friction in places that once felt natural.

That's why product managers need to pause. Not just to evaluate whether something can be built, but whether it belongs. Does it complete the user's experience? Does it connect meaningfully to the rest of the platform?

The goal isn't just to deliver functionality. It's to ensure everything in the system makes more sense because it's there.

Lu doesn't just train the bench press. He trains the setup, the position of his feet, his breathing under load,

and his recovery. Strength alone doesn't make a champion. What matters is the integration of every part.

Product should be no different.

A new feature can introduce more problems than it solves if it isn't thought through holistically. It can expose what's missing more than it reveals what's new. PMs need to take a step back and evaluate not just the capability, but the completeness of the capability across the entire product suite. That completeness is what creates trust.

One of the clearest illustrations of this lives far from software. It lives in a hospital.

At Great Ormond Street Hospital in London, a pediatric cardiac surgery team noticed something troubling. Their operations were world-class, but the transition from surgery to intensive care (the most critical moment) was prone to error. Equipment delays, missed steps, unclear roles. Despite skilled professionals and excellent tools, the system itself wasn't working.

To solve it, they looked beyond healthcare. They studied the Ferrari Formula 1 pit crew.

In an F1 race, a pit stop lasts seconds. Tires change, fuel flows, adjustments happen. All with zero confusion. The

choreography is clear. Roles are defined. Everyone knows their job, and every motion supports the next.

The hospital team brought in Ferrari's pit crew to analyze their process. Inspired by what they saw, the team adopted similar principles. Clear roles. Repeatable sequences. Coordinated language. They didn't add more people or machines. They built better integration.

The result? Fewer errors. Faster transitions. Stronger outcomes. Not because they moved faster, but because they moved together.

That kind of completeness is rare but unmistakable.

Fred Rogers understood it deeply. His television program wasn't flashy or fast. It was thoughtful, quiet, and purposeful. From the music to the pacing, from the way he entered the room to the way he asked questions, every detail was considered. Nothing was accidental. Everything contributed to the experience of a child feeling seen, safe, and understood.

Mr. Rogers didn't ask how to entertain children. He asked what they needed. He aligned his words, his tone, and his format with the emotional weight of that need. He didn't simplify to the point of distortion. He clarified. He built trust by being consistent. (And he built it in Pittsburgh).

Completeness is not a checklist. It is a standard. You feel it in a product that anticipates what you need. You see it in a lifter whose movement is so fluid it barely looks like effort. You experience it in a team that moves with confidence because nothing is missing.

But completeness doesn't begin with execution. It begins with empathy.

The best teams do not just ask what they can build. They ask what someone is trying to accomplish. They understand the pressure, the environment, and the mission that lives outside the interface. They recognize that a feature is not just code. It is part of a system meant to help someone do something that matters.

Empathy isn't softness. It's clarity. It's knowing what to say no to. It's designing for the full context of the user's day, not just the ideal case in a wireframe. It's building for the hard path as well as the happy path.

Mr. Rogers didn't deliver entertainment. He delivered care. He didn't just talk. He listened.

That's what the best products do.

They don't overwhelm. They understand. They don't compete for attention. They earn trust.

Everything built in. Nothing in the way.

That is what makes a system strong. That is what people come back for. Not because it does everything, but because it does the right things - completely.

You're Just Getting Warmed Up

The reps don't stop here.

New chapters are already in the works — deeper lessons from the gym, the product trenches, and the quiet moments in between.

More training logs are coming soon.

Check for updates:

missionbuilt.io

Follow along for sneak peeks and reflections:

bsky.app/profile/missionbuilt.bsky.social

Further Reading

Log 1 - Mission Before Metrics

Case Study: Metrics Over Mission

One Bullet Away: The Making of a Marine Officer

https://en.wikipedia.org/wiki/One_Bullet_Away

Memoir by Nathaniel Fick chronicling his journey from Dartmouth student to Recon Marine officer, revealing the tension between military ideals and battlefield reality.

Cyberpunk 2077 Launch & CD Projekt Red

<https://www.forbes.com/sites/paultassi/2020/12/18/cdpr-stock-drops-after-buggy-messy-cyberpunk-2077-launch/>

A case study in date-driven shipping that sacrificed long-term trust for short-term metrics.

What Really Brought Down the Boeing 737 Max?

<https://www.nytimes.com/2022/10/22/business/boeing-737-max.html>

How engineering culture and metric pressures at Boeing overtook the company's safety-driven mission — with catastrophic results.

Individual Drift: Strength and Misalignment

Hafthor Björnsson on Pec Injury

<https://www.esports.com/bodybuilding-news-probably-my-biggest-mistake-months-after-a-nasty-injury-game-of-thrones-star-makes-bold-confession/>

A personal reflection on chasing numbers too hard — and paying the price with injury.

Hafthor Björnsson Injury Coverage

https://en.as.com/other_sports/game-of-thrones-star-hafthor-bjornsson-suffers-nasty-injury-n/

News coverage reinforcing the story's cautionary angle.

Mission-Aligned Decision-Making

Simone Biles & Mental Health

<https://people.com/everything/simone-biles-has-said-about-mental-health-7561522>

An athlete choosing alignment over achievement — and sparking a global conversation.

Purpose-Driven Brand Behavior

Don't Buy This Jacket – Patagonia

<https://www.patagonia.com/stories/dont-buy-this-jacket/story-18615.html>

Iconic ad campaign where brand values overrode short-term sales incentives.

Research: Purpose & Retention

The Business Case for Purpose

<https://hbr.org/2015/12/the-business-case-for-purpose>

Purpose-driven companies see higher retention, satisfaction, and performance.

Purpose at Work – Imperative & NYU Study (2016)

<https://www.imperative.com/research>

54% of purpose-driven employees are more likely to stay 5+ years.

Purpose: Shifting from Why to How

<https://www.mckinsey.com/business-functions/people-and-organizational-performance/our-insights/purpose-shifting-from-why-to-how>

A roadmap for operationalizing purpose across the org.

Improving Employee Retention

<https://guidehouse.com/insights/healthcare/2023/improving-employee-retention>

Research-based strategies for retention, highlighting mission alignment and culture.

Benevity Talent Retention Study

<https://benevity.com/resources/reports/talent-retention-study>

Mission-driven workers are significantly more likely to stay long-term.

Rituals, Adaptation & Innovation

Stefi Cohen's Hybrid Methodology

<https://www.hybridperformancemethod.com/blog/are-isolation-movements-better-for-muscle-growth>

An elite athlete's fusion of structure and adaptation in strength programming.

Agile Retrospectives

<https://www.agilealliance.org/agile101/agile-retrospectives/>

A guide to turning recurring team reviews into meaningful improvement rituals.

Atlassian ShipIt (Hack Week)

<https://www.atlassian.com/company/team-playbook/plays/>

shipit

A structured ritual encouraging innovation through autonomy and creativity.

Spotify Hack Culture

<https://blog.crisp.se/wp-content/uploads/2011/11/SpotifyScaling.pdf>

How Spotify built agility and creativity into team rituals — not rigid processes.

Strategic Framing

John Boyd's OODA Loop

https://en.wikipedia.org/wiki/OODA_loop

Military strategy model focused on Observation, Orientation, Decision, and Action. Useful for product and performance under uncertainty.

Log 2 - Built Through Reps

The Hidden Grind Behind Success

No Filter: The Inside Story of Instagram

<https://en.wikipedia.org/wiki/>

[No_Filter:_The_Inside_Story_of_Instagram](https://en.wikipedia.org/wiki/No_Filter:_The_Inside_Story_of_Instagram)

A detailed, behind-the-scenes look at Instagram's transformation from Burbn to a cultural phenomenon.

How Steve Jobs Faked His Way Through Unveiling the iPhone

<https://nymag.com/intelligencer/2023/01/how-steve-jobs->

[faked-the-iphone-keynote.html](#)

The legendary 2007 keynote was held together with last-minute fixes — a reminder that polish often hides process.

Instagram: Early History and Pivot from Burbn

<https://en.wikipedia.org/wiki/Instagram#History>

A short overview of Instagram's shift from a check-in app to a global visual platform.

Lifting Progress & Plateaus

Overcoming Strength Training Plateaus

<https://www.ironmaster.com/blog/overcoming-strength-training-plateaus/>

Why strength stalls, how to adjust volume, intensity, and variation to keep progressing.

MV, MEV, MAV, MRV Explained

<https://renaissanceperiodization.com/expert-advice/how-much-should-you-train>

A breakdown of effective training volume concepts, including minimums, maximums, and overreaching thresholds.

Systems, Habits, and Iteration

Atomic Habits

<https://jamesclear.com/atomic-habits>

A guide to building systems that support long-term growth:
“You do not rise to the level of your goals. You fall to the level of your systems.”

The Compound Effect

<https://www.darrenhardy.com/books/compound-effect>

Details how small actions done consistently lead to

exponential outcomes — a key insight for boring but high-leverage work.

Grit: The Power of Passion and Perseverance

<https://angeladuckworth.com/grit-book/>

Explores how consistency beats intensity, and why sticking with it matters more than raw talent.

The Medici Effect

<https://phoscreative.com/big-ideas/the-medici-effect/>

On finding creative inspiration in unlikely intersections — a key idea for hybrid thinkers and builders.

Product Development and Iterative Progress

Failing Fast: Why It's Essential for Entrepreneurs

<https://online.hbs.edu/blog/post/failing-fast>

A reminder that iteration is how we learn — not a sign of failure, but of refinement.

Basecamp's Shape Up Method

<https://basecamp.com/shapeup>

An approach to product work that values thoughtful iteration, foundational improvements, and meaningful user feedback.

Talking to Humans

<https://talkingtohumans.com/>

A practical guide to rediscovering momentum by connecting directly with your users — and learning what really matters.

Is High Quality Software Worth the Cost?

<https://martinfowler.com/articles/is-quality-worth-cost.html>

Argues that invisible work like refactoring and testing pays off in speed and stability — a product parallel to mobility work and back-off sets in lifting.

Log 3 - Rituals Over Rules

Powerlifting Rituals & Adaptive Training

Chalk Up: Rituals That Prepare the Mind and Body

<https://www.elitefts.com/education/pre-workout-ritual-instructions>

Explores how pre-lift rituals like chalking hands and visualization prepare athletes mentally and physically.

Stefi Cohen's Hybrid Training Philosophy

<https://www.hybridperformancemethod.com/blog/are-isolation-movements-better-for-muscle-growth>

Discusses balancing compound and isolation movements, emphasizing adaptability in training.

Autoregulation in Strength Training

<https://barbend.com/autoregulation/>

Details how lifters adjust training intensity based on daily readiness, promoting flexibility within structured programs.

CNS Fatigue: Symptoms and Recovery Strategies

<https://www.strongerbyscience.com/central-fatigue-qa/>

Addresses central nervous system fatigue and strategies for recovery, highlighting the importance of listening to one's body.

Product Rituals, Burnout & Team Evolution

What Is a Stand-Up Meeting?

<https://www.atlassian.com/agile/standups>

Provides guidance on conducting effective stand-up meetings to maintain team alignment.

Leadership @ Elastic: Distributed for the Better

<https://www.elastic.co/blog/distributed-engineering-culture-at-elastic>

Discusses Elastic's approach to distributed work and maintaining engineering culture.

How to Collaborate Effectively If Your Team Is Remote

<https://hbr.org/2021/02/how-to-collaborate-effectively-if-your-team-is-remote>

Evidence-based strategies for improving communication and trust across distributed teams — reinforcing the importance of intentional, adaptive rituals in asynchronous environments.

Shape Up: Ship Work That Matters

<https://basecamp.com/shapeup>

Introduces a methodology that emphasizes flexible work cycles over rigid sprints.

Understanding Developer Burnout

<https://devskiller.com/developer-burnout-guide/>

Breaks down the causes, symptoms, and strategies for preventing developer burnout — with a focus on workload balance, autonomy, and the importance of adaptive team rituals.

Organizational Culture & Ritual Evolution

The Hidden Power of Workplace Rituals

<https://hbr.org/2020/05/the-hidden-power-of-workplace-rituals>

Explores how intentional rituals in the workplace can strengthen psychological safety, enhance purpose, and boost performance.

Crunch Time & Anti-Patterns in Product Culture

Crunch Culture: How Game Development Becomes Toxic

<https://reporter.rit.edu/tech/crunch-culture-how-game-development-becomes-toxic>

Investigates the detrimental effects of excessive overtime in the gaming industry, highlighting how passion-driven projects can lead to burnout and decreased quality.

Red Dead Redemption 2 and Rockstar's Overtime Problem

<https://www.wired.com/story/rockstar-red-dead-redemption-2-crunch-time/>

WIRED explores how extended overtime, driven by passion and pressure, can lead to long-term burnout and erode product quality.

Product Managers as Advocates & Shields

How Product Leaders Can Learn to Say No

<https://www.kennorton.com/essays/no.html>

Ken Norton, former Google PM and GV partner, explains how strong product leaders protect focus by saying “no,” even when pressured by executives.

Log 4 - Feedback Is A Superpower

Coaching & Training Models

Stefi Cohen’s Coaching Evolution

<https://www.hybridperformancemethod.com/blog/are-missionbuilt.io>

isolation-movements-better-for-muscle-growth

Explores Stefi Cohen's evolving approach to training, blending isolation and compound movements, and emphasizing adaptability—a model of feedback-driven progression.

Motor Learning and Coaching Cues

<https://www.drivelinebaseball.com/2017/02/coaching-hitting-mechanics-part-2-application/>

Outlines effective coaching cues in baseball, including directional visual prompts like “turn the flashlight,” to support motor learning through immediate feedback.

RPE and RIR in Strength Training

<https://www.strongerbyscience.com/reps-in-reserve/>

Explains how subjective measures like Rate of Perceived Exertion (RPE) and Reps in Reserve (RIR) can support athlete self-assessment and adaptive programming.

Product Feedback & Strategy

Slack's Origins and Internal Tool Pivot

<https://www.wired.com/2014/08/the-most-fascinating-profile-youll-ever-read-about-a-guy-and-his-boring-startup>

Details how Slack emerged from an internal tool built during game development and refined through continuous team feedback.

Slack's Product Philosophy

<https://review.firstround.com/from-0-to-1b-slacks-founder-shares-their-epic-launch-strategy>

Reveals how early internal use and tight feedback loops shaped Slack's design, long before public launch.

Product Managers Must Talk to Users

<https://www.svpg.com/discovery-feedback>

Marty Cagan explains why user conversations are not optional for PMs, and how intentional feedback is essential to product leadership.

Elastic's Acquisition of Endgame

<https://techcrunch.com/2019/10/15/elastic-adds-endpoint-security-to-its-expanding-toolset/>

Provides context on Elastic's acquisition of Endgame and its feedback-driven product refinement through internal usage and partner design cycles.

Duolingo's Onboarding and Retention Strategies

<https://www.redfast.com/news/how-duolingo-modern-onboarding-drives-user-retention>

Highlights how Duolingo's personalized UX and micro-feedback systems helped increase user retention.

Duolingo's Feedback-Driven UX

<https://blog.duolingo.com/duologues-design-conversations/>

Details how Duolingo designs micro-interactions to offer encouragement and real-time corrections.

Lovable AI and Creative Iteration

<https://lovable.dev/blog/uotthohack-umar-app-development>

Showcases how Lovable's AI tooling supports creative feedback loops in rapid prototyping and design refinement.

Figma's Culture of Real-Time Feedback

<https://www.figma.com/blog/inside-figma-building-a-more-collaborative-design-process/>

Describes Figma's approach to low-friction, high-frequency design feedback through built-in tools.

Misused Metrics & Feedback Failure

Fitbit Sleep Score Anxiety

<https://www.washingtonpost.com/business/2020/02/14/sleep-wellness-employer-oura/>

Explores how a well-intentioned feedback feature from Fitbit led to user anxiety and obsessive tracking.

Target Canada Collapse

<https://canadianbusiness.com/ideas/the-last-days-of-target-canada/>

Examines how Target's failed expansion ignored key operational feedback and misread its own metrics.

The Problem with Listening to Metrics

<https://hbr.org/2022/07/a-better-way-to-put-your-data-to-work>

Analyzes how over-reliance on flawed or misinterpreted data leads to poor decisions and distorted strategy.

Innovation & Creative Feedback

The Medici Effect

<https://www.fransjohansson.com/books>

Frans Johansson's framework for intersectional innovation, emphasizing the power of early, diverse feedback.

Log 5 - Progress Isn't Pretty

Case Studies & Product Examples

Juggernaut Training Systems (JTS)

<https://www.jtsstrength.com>

Founded by Chad Wesley Smith, JTS emphasizes structured training, periodization, and adaptive feedback.

JuggernautAI App

<https://www.juggernautai.app>

An intelligent lifting coach that adapts programming based on lifter input, fatigue, and performance.

Endgame Security Platform

<https://www.elastic.co/about/press/elastic-completes-the-acquisition-of-endgame-a-leader-in-endpoint-protection>

Elastic acquired Endgame in 2019; product evolution included scale-focused rearchitecture and user-centric iteration.

Elastic SIEM Launch Blog

<https://www.elastic.co/blog/introducing-elastic-siem>

Elastic's 2019 announcement of their SIEM solution marks the beginning of iterative product improvement.

Individual Stories

Hafthor Björnsson on Pec Injury

<https://www.menshealth.com/fitness/a63973647/hafthor-bjornsson-elephant-bar-deadlift-record-attempt/>

Björnsson reflects on his injury and emphasizes the importance of smarter, safer training.

Stefi Cohen's Hybrid Method

<https://www.hybridperformancemethod.com/about>

Dr. Stefi Cohen champions hybrid training for power, physique, and longevity.

Philosophy & Models

Malcolm Gladwell's 10,000-Hour Rule

<https://www.newyorker.com/magazine/2008/10/20/late-bloomers-malcolm-gladwell>

Gladwell introduces the concept that achieving mastery in a field often requires approximately 10,000 hours of deliberate practice, emphasizing the role of sustained effort over innate talent.

Perfect Practice Concept

<https://www.psychologytoday.com/us/blog/kluge/201202/will-10000-hours-practice-make-you-expert-anything>

Reinforces that “perfect practice” — not rote repetition — is essential for improvement.

Industry Events

CrowdStrike Global Outage (2024)

<https://www.theguardian.com/technology/2024/sep/24/crowdstrike-outage-microsoft-apology>

Details the real-world impact of a flawed update, validating the importance of safeguards in product delivery pipelines.

Log 6 - The Mission Demands Recovery

Recovery in Sport & Strength

“Hard Work Pays Off” – Mat Fraser Interview

<https://www.youtube.com/watch?v=TLU-Qy0Ireg>

Fraser discusses how disciplined recovery routines, including sleep tracking and structured deloads, fueled his dominance in CrossFit.

Why We Sleep – Matthew Walker

<https://www.amazon.com/dp/1501144316>

A groundbreaking look at how sleep affects memory, learning, performance, and long-term health — essential reading for understanding recovery.

Rest: Why You Get More Done When You Work Less – Alex Soojung-Kim Pang

<https://www.amazon.com/dp/1541617169>

Makes the case for deliberate rest as a strategic advantage in both creative and high-performance fields.

Recovery in Product Work

Netflix Culture Deck

<https://www.slideshare.net/reed2001/culture-1798664>

Outlines Netflix's philosophy on performance, feedback, and the importance of strategic reflection like the “Keeper Test.”

Burnout, Boundaries, and Culture Change

“Work-From-Home Burnout & Zoom Fatigue is a Lot More Complex Than You Think” – Forbes

<https://www.forbes.com/sites/glebtspursky/2023/04/20/work-from-home-burnout-and-zoom-fatigue-is-a-lot-more-complex-than-you-think/>

Explores how remote work blurred boundaries and introduced hidden stressors that accelerated burnout.

“After the COVID-19 Pandemic, Do We Go Back to the Noxious ‘Hustle Culture’?” – Forbes

<https://www.forbes.com/sites/roomykhhan/2021/12/30/after-the-covid-19-pandemic-do-we-go-back-to-the-noxious-hustle-culture/>

Reflects on shifting cultural attitudes toward grind mentality in a post-pandemic world.

“Zoom Fatigue: Solutions” – Wikipedia Summary

https://en.wikipedia.org/wiki/Zoom_fatigue#Solutions

Summarizes research-based strategies to reduce cognitive strain from video conferencing.

Case Studies in Missed Recovery

NASA Challenger Disaster – Rogers Commission Report

https://sma.nasa.gov/SignificantIncidents/assets/rogers_commission_report.pdf

The official postmortem detailing how ignored warnings and organizational pressure led to one of NASA's most tragic and preventable failures.

Evernote Lost Its CTO, CFO, CPO, and HR Head – TechCrunch

<https://techcrunch.com/2018/09/04/evernote-lost-its-cto-cfo-cpo-and-hr-head-in-the-last-month-as-it-eyes-another-fundraise/>

A cautionary tale of how feature sprawl, leadership turnover, and a lack of product reflection led to long-term decline.

Log 7 - Train the Engine, Not Just the Output

Capacity Over Output

General Physical Preparedness: What It Is, Why You Need It

<https://www.crossfit.com/essentials/general-physical-preparedness>

A foundational overview of GPP and how broad, base-building capacity improves performance and resilience.

The Need for GPP Training

<https://www.westside-barbell.com/blogs/the-blog/the-need-for-g-p-p-training>

Louie Simmons outlines how general conditioning supports elite strength development — echoing the engine-building work beneath visible gains.

GPP for Lifters

<https://www.barbellmedicine.com/blog/general-physical-preparedness-gpp-training-for-lifters/>

A modern, practical breakdown of how strength athletes can implement GPP into their training without sacrificing progress.

Resilience Under Load

Product Leadership Is Hard

<https://www.svpg.com/product-leadership-is-hard>

Marty Cagan on the hidden weight of product leadership — and how the boring, essential work often defines long-term success.

Why Every Leader Needs Mental Toughness

<https://www.entrepreneur.com/leadership/why-every-leader-needs-mental-toughness/368965>

A concise case for mental strength as a core leadership trait — especially under pressure.

Cognitive Resilience in Military Personnel

<https://pubmed.ncbi.nlm.nih.gov/35369170/>

A clinical look at how military-style stress inoculation (like missionbuilt.io

SERE) builds internal systems that hold under psychological load.

What Powers You

How Superhuman Built an Engine to Find Product-Market Fit

<https://review.firstround.com/how-superhuman-built-an-engine-to-find-product-market-fit>

Rahul Vohra's data-driven approach to measuring purpose, alignment, and user obsession inside a product team.

The Product Frameworks Behind Superhuman

<https://www.nfx.com/post/superhuman-product-frameworks>

A deeper dive into Superhuman's obsessive product strategy — built on clarity, delight, and speed.

Empathetic and Kind Product Leadership

<https://www.mindtheproduct.com/empathetic-and-kind-product-leadership/>

Why purpose matters not just in mission statements, but in how leaders show up for their teams and users.

Log 8 - Decisions Are Made Under Load

Elastic's Response to Log4j

<https://www.elastic.co/blog/log4j2-vulnerability-what-to-know-security-vulnerability-learn-more-elastic-support>

Elastic's technical response and guidance following the Log4j2 vulnerability, including platform protections and detection content.

Investigate Log4Shell Exploits with Elastic

<https://www.elastic.co/blog/investigate-log4shell-exploits-with-elastic-security-and-observability>

A hands-on look at how Elastic's observability and security tooling supported real-world threat hunting after Log4j.

Detecting Log4j Exploits with Elastic Security Labs

<https://www.elastic.co/security-labs/detecting-log4j2-with-elastic-security>

Elastic's detailed approach to detecting CVE-2021-44228 exploitation using custom rules and threat intelligence.

Chaos Engineering at Netflix

<https://medium.com/netflix-techblog/the-netflix-simian-army-16e57fbab116>

The original post detailing Chaos Monkey and the Simian Army—Netflix's proactive approach to testing for failure in production.

CrossFit: Preparing for the Unknown and Unknowable

<https://games.crossfit.com/workouts/open/2023>

The official CrossFit Open page, reflecting the sport's philosophy of surprise as a test of comprehensive readiness.

Models & Frameworks

John Boyd and the OODA Loop

<https://psychsafety.co.uk/john-boyd-and-the-ooda-loop/>

A plain-language overview of Boyd's Observe–Orient–Decide–Act model and how it applies to business and leadership under pressure.

Netflix uses the OODA loop to stay ahead of the competition

<https://www.techtarget.com/searchcio/news/2240203084/Netflix-uses-the-OODA-loop-to-stay-ahead-of-the-competition>

Netflix's survival method of choice: Observe, Orient, Decide and Act.

Research & Analysis

CSRB Report on Log4j

https://www.cisa.gov/sites/default/files/publications/CSRB-Report-on-Log4J-11-2022_508.pdf

Official U.S. government review of the December 2021 Log4j vulnerability response, including lessons learned and mitigation gaps.

How Tech Giants Built Resilient Systems with Chaos Engineering

<https://sdtimes.com/softwaredev/how-tech-giants-like-netflix-built-resilient-systems-with-chaos-engineering/>

An industry-wide snapshot of how chaos engineering uncovered critical weaknesses and improved production stability.

Training for the ‘Unknown and Unknowable’: CrossFit and Evangelical Temporality

https://www.researchgate.net/publication/337190829_Training_for_the_Unknown_and_Unknowable_CrossFit_and_Evangelical_Temporality

Academic perspective on CrossFit’s ritual design, unpredictability, and how stress is embedded into practice as preparation.

Log 9 - Ship It Like You Show Up

The Legacy of Lü Xiaojun: From Champion to Icon

<https://luxiaojun.com/blogs/blogs/the-legacy-of-lu-xiaojun-from-champion-to-icon>

Official blog post detailing how Lü's grip, breath, posture, and movement combine into a seamless, integrated elite lifting system.

Great Ormond Street Hospital for Children: Ferrari's Formula One Handovers

<https://asq.org/quality-resources/articles/case-studies/great-ormond-street-hospital-for-children-ferraris-formula-one-handovers-and-handovers-from-surgery-to-intensive-care?id=fbc699af11d04980ade06f409a5d6f98>

ASQ case study on how a pediatric cardiac team adopted pit-stop choreography to reduce handoff errors and improve ICU transitions.

Great Ormond Street Hospital Learns from Ferrari F1 Pitstop

<https://outsideinlens.substack.com/p/great-ormand-hospital-learns-from>

Substack narrative on using race-style coordination to streamline critical surgical handovers.

What Mr. Rogers Can Teach Us About Empathy and Why It Matters

<https://www.ei-magazine.com/post/what-mr-rogers-can-teach-us-about-empathy-and-why-it-matters>

Emotional Intelligence Magazine explores Fred Rogers' mastery of empathic listening and emotional depth.

Mindsight and Kindness: Emotions are “Mentionable and Manageable”

<https://drdansiegel.com/emotions-are-mentionable-and-manageable-the-mindsight-and-kindness-of-mr-rogers-brain/>

Neuroscientist Dan Siegel examines Rogers' emotional clarity and how it shaped his compassionate communication.

Seven Lessons from Mister Rogers That Can Help Americans Be Neighbors Again

https://greatergood.berkeley.edu/article/item/seven_lessons_from_mister_rogers_that_can_help_americans_be_neighbors_again

A Greater Good article on how Fred Rogers modeled calm communication, empathy, and social cohesion for over 30 years.

Introducing Project Paper Cuts

<https://github.blog/news-insights/product-news/announcing-paper-cuts/>

GitHub engineers describe how fixing small UX issues—or “papercuts”—improves experience and builds trust.

Keeping Customers Streaming — The Centralized Site Reliability Practice at Netflix

<https://netflixtechblog.com/keeping-customers-streaming-the-centralized-site-reliability-practice-at-netflix-205cc37aa9fb>

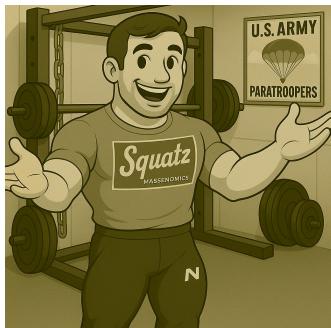
Netflix CORE team shares how owning reliability across the stack ensures consistent service and aligns with mission-first shipping.

Mistakes and Discoveries While Cultivating Ownership

<https://www.infoq.com/presentations/netflix-five-level-ownership/>

Discussion on Netflix's five-level ownership model—from demonstration to vision—and its impact on quality and accountability.

About the Author



Mike Nichols builds products and lifts heavy things — both in service of something bigger than himself.

He began his journey as an Airborne intelligence sergeant in the U.S. Army, where he learned the value of discipline, clarity under pressure, and showing up for the team. Those lessons carried into civilian life, where he spent over two decades in cybersecurity — from analyst to product leader — helping teams ship tools that protect people and infrastructure around the world.

But this book isn't about titles or timelines. It's about the patterns that repeat across every meaningful pursuit: showing up when it's hard, lifting others when you can, and staying true to the mission even when no one's watching.

Mike trains in a garage gym, leads a product team, raises incredible kids, and does his best to live what he writes — imperfectly, but with intention.

You can find him at missionbuilt.io, where this work continues, one rep at a time.

Acknowledgments

This book — like any mission worth building — wasn't accomplished alone.

To the *Mission Built* community: your reps, your rituals, and your resilience inspired every chapter. Whether you're building strength under the bar or building trust in a product, your dedication is proof that craft still matters.

To the teams at Endgame, Elastic, and every engineer, designer, PM, user, and customer who ever cared enough to raise a hand, file a bug, or challenge the roadmap — thank you. You made me better.

To my fellow veterans: your service and sacrifice are the blueprint for what it means to show up with integrity. This book is a tribute to the discipline we carry forward.

To my kids: you remind me that strength isn't just power — it's presence, patience, and persistence. I hope you see yourselves in these pages one day and know that everything I build, I build with you in mind.

To my beyond: thank you for seeing me — not just in the spotlight, but in the shadows. For standing beside me in the chaos and the quiet. You are the home I never have to return from, because you are with me in every rep, every word, every breath of this mission.

And to those quietly doing the hard work every day — refining their process, lifting others, staying true to the mission — you're already part of this story.

Stay strong. Stay focused. Stay built for more.

— Mike

Embracing the Tools That Accelerate the Work

This book was written by a human — with the help of machines, mentors, and mission-driven design.

Many of the ideas and reflections in *Mission Built* were forged through years of reps in the gym, in product rooms, and in life. But bringing those lessons to the page — and into your hands — was made possible with support from a few remarkable tools:

- **OpenAI's ChatGPT** helped organize thoughts, iterate on structure, and edit raw field notes into something more readable, while preserving voice and intent.
- **Sora** was used to generate visual representations of the book's themes — from garage gyms to home offices — bringing energy and imagination to the creative process.
- **Lovable** inspired much of the visual language and layout structure, especially in how early decks and storyboards translated the mission into motion.

Every word, story, and metaphor came from real reps and real reflection. But just like good gear and strong teammates, these tools helped bring it all together with more clarity and care.

The ideas are mine. The execution was supported.

— Mike