THE
ENOUGH BUCKET

A FIELD GUIDE FOR LEARNING DESIGNERS
SUFFERING FROM ENOUGH BUCKET DYSMORPHIA

STEVE CORNEY

CONTENTS

PROLOGUE:
YOU'RE NOT BROKEN.
YOU'VE JUST GOT EBD.

S o there I was, 2:47 am on a Thursday, explaining to the voices in my head why the branching scenario needed seventeen different paths because "what if someone works part-time in regional Tasmania and only uses Internet Explorer on Wednesdays?"

The voices were not impressed.

That's when I knew I had a problem.

Not the voices thing. The other thing.

The fact that I'd just spent six hours building a decision tree for a module about updating your email signature. Six hours. For email signatures.

My coffee had gone cold. My reflection in the monitor looked increasingly concerned. And I was still there, hunched over my laptop like Gollum with a *Captivate* licence, muttering about "comprehensive user pathways" and "inclusive design thinking."

That's EBD for you. It sneaks up on you when you're not looking.

One minute, you're building a simple training module. The next minute, you're explaining to yourself why learners need to understand the historical context of corporate email etiquette before they can change their signature from "Sent from my iPhone" to something more professional.

Welcome to my intervention. And yours.

WHAT THE HELL IS EBD?

EBD. Enough Bucket Dysmorphia.

You've got it. I've got it. Every learning designer who's ever given a shit has got it.

You're not broken. You're not bad at your job. You're just like

every other designer who started with good intentions and ended up building digital monuments to anxiety.

Your Enough Bucket should be teacup-sized. Neat. Contained. Just right.

But EBD makes you see your bucket as smaller than it actually is. You keep pouring more in, thinking there's still room. Meanwhile, your bucket has swollen to swimming pool proportions, and you can't see it.

EBD is a form of reverse dysmorphia. You've already created a learning banquet, but in the mirror, all you see is an empty plate.

It starts with "Maybe I'll just add…"

It ends with a 14-module onboarding that no one finishes and a PDF no one downloads.

WHY THIS BOOK EXISTS

This isn't meant to bring shame to your already fried brain. It is not here to tell you to be "more strategic" or "less emotional" or to stop caring so much about your work.

This book is a mirror—a brutally honest one.

It's a set of road signs that indicate you've gone too far. A diagnosis guide for when your good intentions turn into learner torture. And a companion for when you need to laugh at your own messes before cleaning them up.

Because here's the truth no one wants to say out loud: **Designing learning makes you weird.**

You start to see learners everywhere. You lose sleep over learning objectives. You build things so full of love, content, interactivity, and stakeholder approval that they become the learning equivalent of an all-you-can-eat buffet… for someone who only wants a sandwich.

THE INTERVENTION STARTS NOW

We're going to laugh at the symptoms. We're going to recognise the patterns. We're going to meet the characters who've been sabotaging our designs from the inside.

You'll meet Gavin, the Stakeholder Super-Spreader. Last-Minute Larry, who turns every project into a crisis. Justin Case, who builds for problems that don't exist. Karen from Compliance, who thinks everything needs a disclaimer.

You'll recognise yourself in all of them. That's the point.

This isn't a manifesto. It's a survival guide. One part instruction, one part confession, one part group therapy session for people who care too much about learning design.

Because if we can't laugh at the shit we've built, we'll never fix it.

Welcome to The Enough Bucket.

Let's drain a little. Together.

HOW TO USE
THIS FIELD GUIDE

L et's be clear. This is NOT a textbook.

If you're here, you probably care too much about learning design. You've built courses so stuffed with good intentions they've turned into digital hoarding situations. You've said "just one more thing" so many times you need a support group.

Welcome. You've got EBD. Enough Bucket Dysmorphia.

Here's how this works:

IF YOU'RE IN CRISIS

(mid-project EBD flare-up)

Skip straight to **Chapter 17: The EBD First Aid Kit**. Apply immediate symptom relief, then come back to understand what the hell just happened to you.

IF YOU'RE NEW TO THE FIELD

Read the first few chapters to understand the symptoms, then take the EBD Assessment before you start your next project. Prevention is better than cure, especially when the cure involves deleting three days of work at 2am.

IF YOU'RE A VETERAN DESIGNER

Jump around. You'll recognise yourself everywhere. Focus on the chapters that make you uncomfortable - those are your trigger symptoms. Use this as a diagnostic tool for your own design habits.

IF YOU MANAGE DESIGNERS

Read **Chapter 2: The Stakeholder Super-Spreader** first. Please stop being Gavin. Then, check out the sections on managing EBD in your team. Your designers are drowning

in good intentions, and you might be holding their heads underwater.

IF YOU'RE A STAKEHOLDER READING THIS

First, thank you for caring enough to understand what we're going through. Second, every "quick suggestion" you add after sign-off feeds the EBD monster. Third, "Phase Two" is not a real thing. We made it up to make you go away.

THE STRUCTURE

Each chapter follows the same pattern:

- **Setting the Scene** (the absurd but so very real reality)
- **Symptom Diagnosis** (what's actually happening)
- **Steve's Epic Fail** (so you know you're not alone)
- **Internal Dialogue** (the lies Steve and we tell ourselves)
- **Reality Check** (what learners actually think)
- **Early Warning Signs** (spot it before it spreads)
- **Recovery Actions** (immediate treatment)

REMEMBER

Your Enough Bucket should be teacup-sized. When it swells to bathtub proportions, and you can't see it, that's dysmorphia. This field guide helps you see clearly again.

Now, let's empty some buckets.

EBD SEVERITY ASSESSMENT

Because denial isn't just a river in Egypt

THE TEST

Rate each statement from 1 (never) to 5 (constantly). Be honest. Your Storyline licence depends on it.

1. I add "just one more thing" to modules after they're "finished"

2. I include content "just in case someone needs it"

3. I struggle to delete work, even when it doesn't fit

4. Stakeholder feedback makes me panic-add rather than strategically respond

5. I rebuild things multiple times because they're "not quite right"

6. I include extra resources, links, and downloads "to be helpful"

7. I create complex interactions when simple would work better

8. I give learners multiple paths when one clear route would suffice

9. I reuse old content without questioning if it's still relevant

10. I agree to unrealistic deadlines then build something kitchen-sink to cope

YOUR EBD DIAGNOSIS

10-20 POINTS: MILD EBD *Manageable with awareness*

Congratulations! You're either new to the field, exceptionally disciplined, or lying to yourself. You show occasional symptoms but haven't fully succumbed to the dark side yet. You still delete things without crying. You can say no to stakeholders without breaking into a cold sweat.

Treatment: Weekly reality checks, peer review, and a sticky note on your monitor that says, "Is this necessary?" You're in the prevention phase. Guard this score with your life.

Prognosis: Excellent, provided you don't get promoted to senior designer or put in charge of stakeholder management.

21–35 POINTS: MODERATE EBD *Needs active management*

You're in the danger zone. You've started adding accordions "for better organisation" and using phrases like "comprehensive coverage" unironically. You feel personally attacked when someone suggests cutting content. You've built at least one module that required its own sitemap.

You're not lost yet, but you can see the cliff edge from here.

Treatment: Daily bucket limits, mandatory feedback filtering, and practising saying "Phase Two" in the mirror. Consider hiring someone whose job it is to delete your work.

Prognosis: Treatable with a commitment to change and possibly therapy.

36–50 POINTS: CHRONIC EBD *Intervention required immediately*

Oh dear. You're deep in the content woods, and you can't see the forest for the flip cards. You've built modules that have their own table of contents. You create backup slides for your backup slides. You've used the phrase "learner journey mapping" in casual conversation.

You're the person who adds a glossary to a module about updating your email signature. You export from *Rise* to Storyline because you need "more control." You have strong opinions about hover states.

Treatment: Immediate intervention, project reconstruction, professional support, and possibly an exorcism. Step away from the authoring tool. Call someone who will tell you the truth.

Prognosis: Serious but recoverable with a commitment to change, daily accountability, and accepting that good enough is actually good enough.

SPECIAL DIAGNOSTIC CATEGORIES

SCORED OVER 50? You either made a mistake, or you're patient zero for a new strain of EBD. Please report to the nearest learning design clinic immediately. Do not pass GO. Do not add another accordion.

SCORED UNDER 10? You're either a sociopath, or you work in compliance where everything really does need seventeen disclaimers. Either way, we're concerned.

ALL 5S? You're having a breakdown. Close this book. Call a friend. Touch grass. The modules will wait.

ALL 1S? You're lying. Nobody is that disciplined. Try again and be honest this time.

WHAT HAPPENS NEXT

If you scored in the Mild range, use this book as prevention. Learn from our mistakes.

If you scored Moderate, this book is your intervention manual. Read it. Apply it. Check in weekly.

If you scored Chronic, this book might save your career. And your sanity. Start with Chapter 17 (Emergency Protocols), then come back to Chapter 1.

Remember: EBD is not a moral failing. It's an occupational hazard. The fact that you care enough to take this assessment means you're already on the path to recovery.

Now stop procrastinating and get back to building something simple that works.

QUICK SYMPTOM CHECKER

When EBD bleeds into real life

PHYSICAL SYMPTOMS

- Compulsive clicking "Save" on authoring tools
- Phantom buzz from Slack notifications about "one more change"
- Stress dreams about accordions that never end
- Inability to close browser tabs with "helpful resources"
- Muscle memory for keyboard shortcuts in software you don't even use
- Eye strain from staring at hover states for too long

EMOTIONAL SYMPTOMS

- Guilt when deleting content
- Panic when someone says "this feels light"
- Pride in module completion times over learning outcomes
- Defensive when anyone suggests cutting content
- Euphoria when discovering a new authoring tool feature
- Existential dread when facing a blank slide

BEHAVIOURAL SYMPTOMS

- Building backup slides for backup slides
- Creating "optional" sections that aren't really optional
- Justifying every design choice with "learners might need this"
- Refreshing LMS analytics obsessively

- Version numbering that looks like software releases (v2.7.3_final_FINAL_steve_stop)
- Talking to yourself about user experience while building

WHEN EBD INVADES YOUR PERSONAL LIFE

At social gatherings:

- Explaining Bloom's taxonomy at BBQs ("Well actually, grilling is more than just remembering - it requires application and synthesis...")
- Critiquing restaurant menus for poor information architecture
- Referring to your holiday plans as "a learning journey"
- Using phrases like "that's a great insight" in casual conversation

At home:

- Creating decision trees for what to watch on Netflix
- Building IKEA furniture and muttering about "poor instructional design"
- Organising your spice rack with learning objectives
- Writing shopping lists with clear success criteria

In relationships:

- Asking your partner for "feedback on your communication strategy"
- Creating branching scenarios for difficult conversations
- Using the phrase "let's circle back on that" during arguments
- Suggesting your relationship needs "better stakeholder alignment"

With your kids:

- Turning bedtime stories into interactive experiences with decision points
- Creating chore charts that look like learning management systems
- Explaining why they can't skip to dessert because "you haven't completed the prerequisites"
- Using learning objectives for family holidays ("By the end of this trip, you will be able to...")

At the grocery store:

- Mentally redesigning the layout for "better user flow"
- Getting frustrated by poor wayfinding signage
- Organising your trolley contents by learning module
- Explaining to checkout staff why their process needs more scaffolding

While driving:

- Critiquing road signs for accessibility compliance
- Creating mental flowcharts for traffic light sequences
- Referring to GPS instructions as "just-in-time learning"
- Getting angry at roundabouts because they're "too many pathways to the same outcome"

RED FLAG SYMPTOMS

If you're experiencing any of these, seek immediate help:

- You've gamified your morning routine with badges and progress bars
- You created a branching scenario to decide what to have for lunch

- You referred to your last breakup as "scope creep in the relationship module"
- You've started using SCORM compliance as a dating criterion
- You explain movies using the ADDIE model
- You've built a learning path for teaching your dog to sit

THE WEEKEND TEST

Healthy designer: Switches off, enjoys hobbies, doesn't think about work

Mild EBD: Occasionally notices poor design in everyday life but moves on

Moderate EBD: Takes photos of bad signage to "use as examples in future training."

Chronic EBD: Spends Saturday rebuilding the local cafe's ordering process in Storyline "just for fun". Fun fact. Steve has actually done this. More than once.

If you ticked more than 3 in any category, keep reading, mate. We've got work to do.

If you recognise yourself in the personal life examples, put down the laptop and slowly back away from the authoring tool. Your family misses you.

CHAPTER

1

KITCHEN SINK SYNDROME

Because you can, so you do

LET'S SET THE SCENE

You're designing onboarding. Simple brief: "Help new starters find the bathroom and log into their computer."

Six weeks later, you've created a 47-module epic with drag-and-drop personality assessments, branching scenarios for "optimal desk placement," and a Vyond animation about your company's history that no one requested.

The bathroom location? Hidden behind a flip card on slide 312. After the accordion about org chart hierarchies.

SYMPTOM DIAGNOSIS

Kitchen Sink Syndrome is the purest EBD symptom. You care about the learner. You can build amazing interactions. So you bloody well do. All of them. At once.

You're like a caffeinated designer with a Storyline licence and no adult supervision. Variables? You've got them. State changes? Seventeen different ones. JavaScript? Why not make the buttons dance?

You want to show off every feature you've mastered while giving learners every possible way to consume content. Flip cards, accordions, drag-and-drop exercises, and SCORM packages that could run a small country.

It starts innocently. You add the basics in *Rise*. Then you think, "This could be more interactive", and export to Storyline. Then, you discover triggers. Then variables. Then, you're building a choose-your-own-adventure experience with more complexity than a NASA launch sequence.

STEVE'S EPIC FAIL

I created an onboarding course that was so packed that it required its own sitemap and a mascot that followed you through each slide. It included Blink-tracking. Because why not?

I believed I was demonstrating "advanced authoring capabilities." Really, I was like a toddler with Legos: just because I could build it didn't mean anyone wanted to live in it.

INTERNAL DIALOGUE

"Learners expect interactive modules now."

"It's not extra, it's immersive."

"The CEO might see this. It has to be impressive."

"If I don't use Storyline, what was the point of learning it?"

"Nobody wants to click 'Next' 12 times."

"Okay, but the animated flowchart IS cool."

"I'll just add one more thing."

REALITY CHECK

"I just wanted to know where the printer is. Why am I dragging office furniture around like I'm playing *The Sims*?"

"Your mascot won't stop talking. I've learned more about this cartoon character than my actual job."

"This feels like someone gave a sugar-rushed kid access to *Captivate* and told them to go nuts."

They didn't appreciate my advanced authoring skills. They just wanted to find the bloody bathroom.

EARLY WARNING SIGNS

- You add interactions because you learned how to build them
- Your *Rise* courses get exported to Storyline "for more functionality"
- You use phrases like "immersive learning experience"
- Your SCORM packages are larger than most people's photo libraries
- You feel proud of your "sophisticated interactions"

RECOVERY ACTIONS

The Tool Test: Before adding any interaction, ask, "Does this serve learning, or does this serve my ego?"

The Basic Bitch Challenge: Build your next module in *Rise*. If you can't, you've got Kitchen Sink Syndrome.

The Hover State Intervention: If you're customising button animations at 2 am, step away from the computer.

Remember: learners don't give a shit about your JavaScript skills. They want clarity, not complexity. Your job is to teach, not to audition for a motion graphics portfolio.

CHAPTER

2

THE STAKEHOLDER
SUPER-SPREADER

Gavin strikes again

LET'S SET THE SCENE

Y ou were doing exceptionally well. You had scoped the
module. You had a clear learning outcome. You had
trimmed the fat and were just one "next" button away
from something that felt right.

Then Gavin provided feedback.

Like a feedback gremlin that multiplies when fed after midnight
- or, worse, after the review round - one suggestion generates
three more. "Could we simply add the policy context? And
perhaps the case study from last year's conference?"

Suddenly, Karen from Compliance appears with "safety
considerations." The CEO's cousin has "just a quick thought"
regarding user experience. Marketing remembers having
slides from that conference. Each gremlin spawns another
gremlin, and your clean design is overrun by a swarm of well-
meaning content destroyers.

SYMPTOM DIAGNOSIS

The Stakeholder Super-Spreader occurs when one person
keeps adding "just one more thing" after you've already built
something functional. Gavin does not intend to ruin your
design. He's just following the SME Playbook: more content
equals more value.

You say yes because you want to be seen as collaborative.
However, Gavin's suggestions are like a virus that spreads
throughout your clean design, infecting other stakeholders
with the desire to contribute. What began as feedback
became an epidemic.

Your elegant 10-minute module becomes a 45-minute
sprawling mess because you couldn't find the antidote to
stakeholder suggestion syndrome.

STEVE'S EPIC FAIL

Gavin once sent me feedback on a "finished" customer service module. My sleek 12-minute module mutated into a 34-minute stakeholder casserole. Everyone added their secret sauce. Nobody could stomach it.

INTERNAL DIALOGUE

"Gavin probably knows what learners need."

"If I push back, they'll think I'm not collaborative."

"It's just one more slide, no big deal."

"Okay, I'll add it... and then maybe trim later."

"Can't say no now, legal's seen it."

"This is what partnership looks like, right?"

REALITY CHECK

"I just wanted to learn how to handle complaints. Why am I reading the entire escalation policy?"

"This feels like Gavin threw up his brain all over my screen."

"I started a 12-minute module, and three hours later, I'm still clicking through optional resources about things I'll never use."

The feedback gremlins had successfully transformed my focused learning into a stakeholder anxiety dumping ground.

EARLY WARNING SIGNS

- Gavin sends "quick thoughts" after sign-off
- You implement suggestions without questioning them
- Other stakeholders start contributing "while we're at it"

- Your modules keep growing after you thought they were done

- You justify bloat with "the SME requested it"

RECOVERY ACTIONS

The Scope Shield: Hold up your original project brief and say, "Phase Two! Phase Two!" until the gremlins retreat to their stakeholder caves.

The Gavin Filter: Ask: "Does this help achieve the original learning outcome?" If no, it goes into the feedback graveyard.

The Evidence Test: "What evidence do we have that learners need this?" Watch the suggestion demons vanish when they can't provide proof.

The Quarantine Protocol: When one gremlin suggests something, don't let it spread to other departments. Contain the outbreak before it becomes an epidemic.

Your job isn't to be Gavin's content dumping ground. It's to protect the learner from the stakeholder zombie apocalypse.

3

PERFECTIONISM IN DISGUISE

The Designer's Masquerade Ball

LET'S SET THE SCENE

I t's 1:37 a.m. In *Captivate*, you are arguing with yourself about the size of the buttons. You have exported the SCORM package five times without changing anything. You're in a Google Doc, trying to rewrite a sentence that you already nailed last week.

You tell yourself it isn't ready. It only needs a few more tweaks. Perhaps another layout pass. Maybe the hover states should be 0.2 seconds faster. Maybe the accordion animation should be ease-in-out rather than ease-in.

Stop it. You're not refining. You're spiralling like Gollum, clutching his cherished hover state and whispering, "Just one more tweak, my precious."

SYMPTOM DIAGNOSIS

Perfectionism in Disguise isn't about getting things right. It's about being scared shitless someone will think you missed something. You're not a perfectionist because you're precise - you're a perfectionist because you're terrified.

It looks like attention to detail. It smells like professionalism. But it's content madness with a business card.

You'll rename the same lesson three different times and hate all of them. You'll spend two days rebuilding a button animation because it's not "as smooth as it could be." You're not building for the learner - you're building monuments to your own ego overlord.

STEVE'S EPIC FAIL

I once spent two days rebuilding a button animation for a mandatory compliance refresher about workplace harassment. Two. Days. On a hover effect.

The learner? A time-poor frontline worker who just needed to tick the box and move on with their life. I poured my soul into a hover effect for someone who just wanted to make it to lunch.

The animation was beautiful. Buttery smooth. Perfect ease curves. The learner never noticed it because they completed the module on their phone while walking to the car park.

INTERNAL DIALOGUE

"If this looks sloppy, they'll think I don't care."

"Great designers obsess over details."

"This hover feels off... maybe by 0.5 seconds."

"People *will* notice. Even if they don't say it."

"It's not ready. One more pass."

"This version *might* be better than v14. Let's check."

REALITY CHECK

"Your course works fine. Why are you still working on it?"

"I can't tell the difference between version 3 and version 8. They're both good."

"You've been 'almost finished' for three weeks. Just publish the bloody thing."

While I was in version twelve, obsessing over kerning (spacing between letters ffs), the learner had moved on. They didn't need perfect. They needed finished.

EARLY WARNING SIGNS

- You use the phrase "just one more pass" repeatedly
- You rebuild the same interaction three different ways
- You export preview versions for yourself to "check"
- You have strong opinions about micro-interactions in compliance training
- You feel physically uncomfortable clicking "publish"

RECOVERY ACTIONS

Set Completion Triggers: Before you start, write down three criteria that mean "this is done." When you hit all three, hands off.

The 80% Rule: If it's 80% right and serves the learner's need, ship it. The remaining 20% is usually just your ego talking.

The Colleague Handoff: Send your "finished" module to someone else for final export. Remove your own access. Seriously.

The Timer Method: Set a 25-minute timer for revisions. When it goes off, you're done.

The cure? Press publish. And then go touch grass. Good enough is where learning starts. Perfect is just fear in a tailored suit.

4

CLUTTER OF CARE

Built with love. Bloated with guilt.

SET THE SCENE

You finish building a module and open it back up to admire your work. What you've actually built is a less "engaging experience" and more digital garage sale.

There's a checklist. There's a dropdown. There's a five-step accordion with links to two policy PDFs and a quote from Brené Brown about vulnerability that you thought was "really powerful" at 2 am on a Tuesday.

You've filled the thing tighter than a carry-on bag packed by someone with trust issues. It's bursting. None of it is bad, but all of it together? It's chaos.

Welcome to the Clutter of Care - like a rogue Care Bear gone postal, smothering learners with rainbow safety nets until they can't breathe.

SYMPTOM DIAGNOSIS

The Clutter of Care is what happens when you love your audience so much you start hoarding content in their honour. You don't want to leave anything out. You don't want to let anyone down.

This isn't Kitchen Sink Syndrome - that's when you throw in features because you can. And it's not Stakeholder Super-Spreader either - Gavin's not making you do this. You're doing this to yourself out of pure, misguided love.

Clutter of Care is when you throw in content because you care. You're not showing off your authoring skills; you're making sure learners have everything they could possibly need, even if they don't know they need it. You've become a one-person content committee, but instead of arguing with stakeholders, you're arguing with your own guilty conscience.

So you include the onboarding checklist, the optional explainer video, the team bios (including Kevin, who no longer works there), the case study that someone in leadership said is "really powerful," and the reflective journaling prompt that you would personally love to do.

You've become an educational hoarder. Every piece of content is precious. Every resource is "just in case someone needs it." Every link is wrapped in good intentions and tied with anxiety ribbons.

STEVE'S EPIC FAIL

I once built a customer service training so full of "helpful extras" that it crashed the LMS twice. Not because of technical issues - but because of the sheer weight of caring.

I curated a soundtrack for customer service excellence because I cared. So. Bloody. Much.

INTERNAL DIALOGUE

"This resource might really help *someone*."

"It's not clutter, it's comprehensive."

"I want them to feel supported. Loved, even."

"It's only optional. They can skip it."

"What if someone needs exactly this one obscure thing?"

"Deleting it feels like I'm letting them down."

"This is what transformational learning really looks like."

REALITY CHECK

"Overwhelming." "Too much." "Made me anxious." "Felt like homework for a subject I didn't sign up for."

One person wrote: "I just wanted to learn how to handle angry customers. Instead, I got a masterclass in everything except that."

My clutter of care had become a haunted house of helpfulness where every door led to another corridor of assistance.

EARLY WARNING SIGNS

- You describe your modules as "resource-rich"
- You have more optional elements than required ones
- You feel guilty removing stakeholder contributions
- You create content "just in case" someone wants to go deeper
- You use phrases like "comprehensive support materials"

RECOVERY ACTIONS

The Marie Kondo Method: Ask each piece of content: "Does this spark learning?" If no, thank it for its service and delete it.

The One-Page Summary: If you can't summarise your module on one page, it's too cluttered.

The Attention Budget: Learners have 20 units of attention. How are you spending them?

The Clarity Champion: Assign someone to be the voice of simplicity in your design process.

Caring isn't the problem. Clutter is. If your love for the learner turns your design into a haunted house of helpfulness, they're not gonna thank you. They're gonna leave. Quietly. Three slides in.
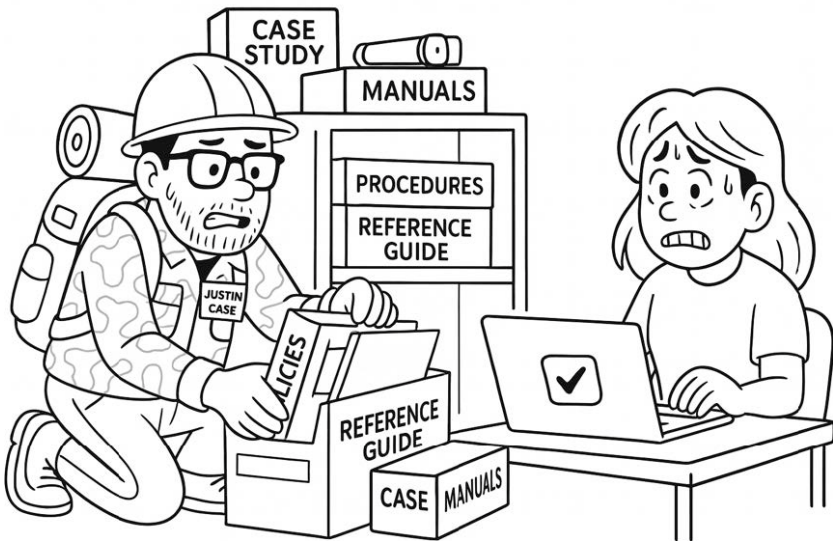
Nothing says "I respect your time" quite like a course that ends before their coffee goes cold.

5

JUST-IN-CASE CREEP

A tragic tale of one man, one dropdown, and the apocalypse that never came

SET THE SCENE

Meet Justin. Justin Case. He's a good bloke. Cares a lot. Real proactive operator. And he's ruining your course. Justin's the voice in your head saying:

- "Let's include the extra policy doc… just in case."
- "Might be good to add a case study from 2011… just in case."
- "Could we chuck in a separate path for regional Tassie staff accessing on dial-up Mozilla Firefox on Windows 98 powered by a generator… just in case?"

You don't know when he arrived. You just woke up one day, and your perfectly focused *Rise* course had been exported to Storyline so you could build 16 different branching scenarios, all because Justin didn't want to leave anyone behind.

Now you're building for ghosts like a doomsday prepper who's convinced the learning apocalypse is coming and only a bunker full of contingency content will save us all.

SYMPTOM DIAGNOSIS

Just-in-case creep is the ultimate EBD symptom. It feels logical. It feels safe. It feels like you're being thorough.

But what you're actually doing is designing from fear. Fear that someone might need it. Fear that someone might complain. Fear that without it, the learner won't "get it."

So, you build for the edge case. Then, the outlier. Then, the hypothetical learner who hasn't been onboarded yet but might be hired next year and speaks fluent PowerPoint but only on Tuesdays.

Justin doesn't just fill your Enough Bucket. He spikes it with panic. It's a survivalist learning design.

STEVE'S EPIC FAIL

My compliance module was done. Then Justin sent me a suggestion. Five days later, I took a breath and realised I'd built a learning time capsule six layers deep.

The best part? I tracked the analytics in my LMS. Zero clicks. Nobody ever clicked it. But it was there, lurking in my SCORM package like digital tumbleweeds. Because Justin told me: "Someone might."

INTERNAL DIALOGUE

"What if someone actually asks about that?"

"Better safe than sorry."

"It's just one more link."

"If I leave it out, I'll regret it."

"Someone in Tasmania *might* need this."

"I mean… no harm in including it, right?"

REALITY CHECK

"I just wanted to learn the new process. Why are there 13 optional resources about edge cases I'll never encounter?"

"This feels like it was built by someone who's never met an actual employee."

"Your 'just in case' sections are longer than your actual content."

I'd built a content doomsday bunker for problems that didn't exist while the real learners were standing outside wondering why the door was locked.

EARLY WARNING SIGNS

- You use the phrase "what if" more than "this will"
- You build features for scenarios that haven't happened yet
- You include content for users who don't exist
- Your modules have more conditional paths than a Choose Your Own Adventure book
- You feel anxious about NOT including something

RECOVERY ACTIONS

The Evidence Audit: For every optional element, ask, "What evidence do I have that someone needs this?"

The Ghost Hunt: Identify content built for "someone who might." If no one has actually asked for it, delete it.

The 80-20 Rule: Build for the 80% of users who have 80% of common needs. The rest can figure it out or ask for help.

The Reality Check: List your actual users. Document real scenarios they face. Build for them, not for Justin's anxiety.

Here's the truth: If someone might... they probably won't. And if they do? They'll ask.

Every "just in case" item is a withdrawal from the learner's attention budget. And you don't have that kind of credit.

6

INTERACTIVITY FEVER

You get a slider! You get a hotspot!
Everyone gets a bloody accordion!

SLIDER!

HOTSPOT!

ACCORDION!

SET THE SCENE

You just mastered a new Articulate feature. Your brain lights up like a slot machine hitting the jackpot. You drag your first interactive element onto the slide, and suddenly, you're Interaction Man, armed with triggers, variables, and the unshakeable belief that static content is dead.

Everything becomes drag-and-drop. Every sentence needs a slider to reveal it. Your "Welcome" screen now sounds like Mario Kart on a ketamine bender, complete with custom animations that would make a Vegas casino jealous.

Welcome to Interactivity Fever - when you become a caffeinated authoring tool virtuoso with no adult supervision and an Evolve licence that's gone to your head.

SYMPTOM DIAGNOSIS

Interactivity Fever is what happens when you discover the shiny features of your authoring tool and decide EVERYTHING needs to be interactive. You're not building learning. You're building a tech demo disguised as education.

You've just learned how to make buttons spin? Suddenly, every button spins. You discovered branching scenarios? Everything becomes a choose-your-own adventure, including the bloody navigation.

You start measuring success by how many clicks something requires instead of how much someone learns. Your SCORM 2004 packages become larger than Netflix downloads, packed with more animations than a Pixar film.

STEVE'S EPIC FAIL

I turned company values into a 29-minute interactive theme park. Learners came for "integrity," left with RSI from excessive drag-and-drop exercises and a deep fear of animated buttons.

There were five values. No one remembered a single one. But I got complimented on the "slickness" of the UX.

INTERNAL DIALOGUE

"Static content is literally violence."

"Clicks equal engagement. Science."

"This isn't training. It's ART."

"I'm not building modules. I'm crafting EXPERIENCES."

"Dragging builds neural pathways."

"Hover states are the future of human connection."

"If it doesn't spin, it's not learning."

"PowerPoint is for peasants."

"Interactivity separates legends from losers."

"Why read when you can DRAG?"

REALITY CHECK

"I just wanted to know the company values. Why am I playing some weird corporate video game?"

"Your interactions are pretty, but I still don't know what I'm supposed to do differently at work."

They weren't learning. They were trapped in an interactive hellscape with no map, no logic, and a quiz that made them drag tiny icons into boxes like it was a Fisher-Price toy for overachievers.

EARLY WARNING SIGNS

- You add interactions because you learned how to build them
- Your modules have more animations than educational content
- You measure engagement by clicks instead of comprehension
- You feel personally offended when someone suggests static content
- You use phrases like "immersive learning experience"

RECOVERY ACTIONS

The Interaction Hierarchy: Ask, "Is this Essential, Helpful, Decorative, or Distracting?" Only use Essential and Helpful.

The Plain Text Test: Remove all interactions. Does the learning still work? If yes, you don't need the bells and whistles.

The Mobile Reality Check: Use your module on a phone with 3G. If it's painful, you've overdone it.

The Learning First Rule: Before adding any interaction, ask, "Does this serve the learning outcome, or does this serve my ego?"

Interactions are like chilli. Use just enough, and it wakes the dish up. Use too much, and people cry and question their life choices.

Your job is to teach, not to audition for a motion graphics portfolio.

CHAPTER

7

THE PATHWAY PARADOX

More paths, more problems

SET THE SCENE

Welcome to the buffet. A learning buffet where everything is available, all the time, with a side of confusion and a complimentary drink of cognitive overload.

You've got three personas. Each has a different goal. Each path has a different scenario built into the Storyline. And each scenario has multiple endings. Each end has a feedback loop, a reflection prompt, a survey, and a link to your intranet from 2009.

You built a maze. A beautiful, complex, logic-jammed monstrosity that learners will absolutely never finish. Half of them pick the wrong path and don't even realise it. They're stuck in what we call a learning cul-de-sac, wandering around your choose-your-own-disaster like visitors to IKEA who gave up and went for coffee instead.

SYMPTOM DIAGNOSIS

The Pathway Paradox is when you mistake complexity for sophistication. You want learners to feel empowered, so you give them seventeen different ways to reach the same bloody destination.

You're not creating choice empowerment. You're creating choice paralysis disguised as learner-centred design. It's the educational equivalent of dropping someone in the middle of Westfield with no map and seventeen different escalators, none of which go where they think they go.

Not all choice is a good choice. And more choice is definitely not a better choice. Give them six different learning paths, and they'll choose none of them.

STEVE'S EPIC FAIL

My leadership course needed a flowchart to survive. Most learners just got lost and left. Some may still be in there.

Actually, if you see Jimmy from Marketing, we haven't actually seen him. Last seen in the LMS report 36 minutes in was his last click.

INTERNAL DIALOGUE

"Different learners need different paths."

"More choice = better experience."

"They'll appreciate the flexibility."

"Netflix has a lot of options, and people love it."

"This is what adaptive learning looks like."

REALITY CHECK

"I spent more time deciding which path to take than actually learning anything."

"This feels like someone built the IKEA of learning modules. Beautiful in theory, exhausting in practice."

"I just wanted to learn how to give feedback. Why are there seventeen different ways to get there?"

I'd built the learning equivalent of a transport interchange designed by someone who'd never caught public transport. Lots of options, no clear direction, and everyone ends up lost.

EARLY WARNING SIGNS

- You use phrases like "choose your own journey" unironically
- Your modules have more decision points than a telco phone tree

- You build different paths for personas that barely differ

- You create options that no one ever chooses

- You feel proud of your "flexible learning architecture"

RECOVERY ACTIONS

The Default Path Strategy: Create one clear, optimal path for 80% of users. Add choices only where they genuinely matter.

The Path Audit: Map every possible route through your module. If there are more than 3, start cutting.

The Choice Justification: For every option, write a one-sentence reason why it exists. If you can't, it goes.

The User Testing Reality Check: Watch someone navigate your choices without your guidance. Count how many times they hesitate.

Learner-centred doesn't mean learner-burdened. You don't need to offer a tasting platter when a single strong dish will do.

If you want to give people power, give them the tools to succeed. Not menus to get lost in.

8

THE ANTI-SIMPLICITY BIAS

Complexity: now available in six confusing formats, none of which you asked for

LET'S SET THE SCENE

You've just built something clean. Something simple. Something that actually works in *Rise.*

Then that sneaky little goblin in your brain whispers: "This looks too easy." "They'll think I didn't work hard enough." "Let's add a layer. Or three."

So you export to Storyline and dress up your learning like a corporate drag show. More layers. More logic. More language. A three-stage accordion filled with conditional content, reflection journaling, and a branching decision tree because someone mentioned "sophisticated learning architecture" at a conference once.

Congratulations. You've just sabotaged your own design in the name of intellectual insecurity.

SYMPTOM DIAGNOSIS

The Anti-Simplicity Bias is when you build something that works perfectly well and then sabotage it because it looks "too basic." You're embarrassed by simplicity. You think simple equals unprofessional.

This isn't Kitchen Sink Syndrome - you're not adding features to show off. It's not Clutter of Care - you're not hoarding resources out of love. It's not an Expert's Paradox - you're not dumping knowledge because you know too much.

This is pure aesthetic insecurity. You take a perfectly functional *Rise* course and export it to Storyline not because you need more functionality but because *Rise* "looks too simple." You replace clear instructions with elaborate frameworks. You swap plain English for jargon soup. You turn a three-step

process into a seven-step methodology because three steps feel "lightweight."

You're not making it better. You're making it look more impressive. There's a difference.

You think if learners breeze through your module, they haven't learned anything. So, you add cognitive speed bumps. Unnecessary friction that makes the learning feel "substantial." Like adding rocks to a backpack to make a hike more challenging.

You couldn't help yourself because deep down, you're terrified someone will think you didn't work hard enough.

STEVE'S EPIC FAIL

What should've been "how to give feedback" turned into a neuroscience thesis wrapped in a 23-question personality quiz.

For people who just wanted to know how to tell someone their work needed improvement.

INTERNAL DIALOGUE

"Simple = amateur."

"If it's too easy, they'll think I didn't do much."

"I need to show the full depth here."

"Learners should be challenged."

"I didn't do all this research for nothing."

REALITY CHECK

"I feel dumber now than when I started. I just wanted to learn how to have difficult conversations. Instead, I got a PhD in conversation theory that I didn't ask for and can't use."

"This feels like it was built by someone who confused comprehensively with complicated."

"I came here to learn a skill. I left with a reading list that would take me three years to get through."

I'd built an intellectual pissing contest disguised as learning. A monument to my own insecurity about whether simple was smart enough.

EARLY WARNING SIGNS

- You use the word "sophisticated" to describe your modules
- You add complexity to prove you understand complexity
- You feel embarrassed when someone says your module looks "clean" or "simple"
- You justify complicated features with "learners need to be challenged"
- You equate comprehensive with complicated

RECOVERY ACTIONS

The One-Sentence Summary: If you can't explain your module in one sentence, it's too complex.

The Child Test: Could a bright 14-year-old understand this? If not, simplify it.

The Unnecessary Element Hunt: Remove one complex feature and see if the learning still works.

The Plain English Challenge: Rewrite one section using only common words.

Here's the secret: Simplicity is earned. It's not laziness. It's not undercooked. It's not "too obvious."

It's the hardest, most grown-up design decision you can make. And the mark of true expertise? Making complex things simple, not simple things complex.

9

THE COMPLETION CONSPIRACY

When the Cult meets the Curse
and your learners become prisoners

LET'S SET THE SCENE

You've built a course so obsessed with tracking completion that learners can't access anything out of order. Not just modules - individual slides. They must spend a minimum of 2 minutes on each screen before the Next button unlocks. If they try to jump ahead, a pop-up appears: "Please review all content before proceeding."

But wait, there's more.

If they spend less than the "optimal time" on a section, they get redirected to a "supplementary review" module. If they score less than 100% on any quiz, they have to retake not just the quiz but the entire section that led up to it.

Welcome to the Completion Conspiracy, where the Cult (designing for analytics) has joined forces with the Curse (forcing linear consumption) to create the ultimate learning prison. Your LMS has become Alcatraz with SCORM tracking.

SYMPTOM DIAGNOSIS

The Completion Conspiracy happens when two EBD symptoms team up like learning design supervillains. You're not just measuring everything. You're forcing everything. You're not just tracking progress. You're controlling it.

Emperor Analytics has convinced Darth Progress Bar that learners can't be trusted with choice. Together, they've built a Death Star of forced sequential completion where every click is monitored, every pause is measured, and escape is impossible.

Your learners aren't students - they're digital hostages clicking through your content jail just to hit 100%.

STEVE'S EPIC FAIL

I built a leadership program that was simultaneously obsessed with completion metrics AND forced linear progression. I built a digital prison where learners served time. Their only crime? Clicking too fast.

INTERNAL DIALOGUE

"If they can skip stuff, how do we know they learned anything?"

"I need to track every click."

"The business needs hard data."

"Mastery requires time on screen."

"Engagement = linear progression, obviously."

REALITY CHECK

"This feels like being trapped in an educational escape room designed by someone who hates me."

"I've been clicking through mandatory content for two hours. I still don't know what I'm supposed to do differently at work."

"Your course doesn't teach anything. It just tests whether I can follow arbitrary rules while being watched."

The conspiracy had turned learning into surveillance. Learners weren't gaining skills - they were serving time in my digital prison while I collected data about their captivity.

EARLY WARNING SIGNS

- You track completion metrics more obsessively than learning outcomes
- Your courses have more checkpoints than airport security

- You feel personally offended when learners skip optional content
- You use words like "mastery" to justify forced linear progression
- Your LMS dashboard has more data than Netflix analytics

RECOVERY ACTIONS

Break Up the Gang: Separate completion tracking from completion forcing. Measure what people do; don't control what they can do.

The Freedom Test: Remove one artificial barrier and see if learning actually suffers. (Spoiler: it won't.)

Focus on Outcomes, Not Consumption: Ask, "Did they change behaviour?" not "Did they click everything?"

Trust the Learner: Design for choice, not compliance. People are smarter than your analytics dashboard thinks they are.

Remember: learners don't need a progress bar to prove they're learning. They need space to actually learn. Every gate you add, every timer you set, every forced sequence you create is another barrier between them and the thing they came to do.

Break the conspiracy. Set your learners free.

10

THE CURSE OF CONDITIONAL LEARNING

Before we cover Topic A, they'll need Topic B. But to get Topic B, they'll need Topic C. And actually... do they even know the alphabet?

LET'S SET THE SCENE

You've got a simple job: teach Jimmy how to safely submit a risk assessment. He's a project lead. He's done this before. He just needs the updated steps.

But then you get hit with The Spiral.

"Well, they'll need to understand the new policy first." "And some background on why the changes were made." "Maybe a bit of legal context?" "Should we include general safety principles?" "Do they even know what a risk is?"

Suddenly, you're building a six-module epic with pre-work, side quests, and a final boss battle called Cognitive Load. Like a doomsday prepper who won't let anyone into the bunker without proving they understand the entire history of civilisation first.

Your simple *Rise* module has become a branching nightmare with more prerequisite gates than a military base.

SYMPTOM DIAGNOSIS

The Curse of Conditional Learning is when you front-load everything learners "need to know first" before they can access what they actually came for. You're terrified they won't understand without ALL the context.

Karen from Compliance doesn't help. She whispers sweet nothings like, "What if they don't understand the regulatory framework?" Suddenly, your simple "How to Submit Expenses" module needs prerequisite courses, including "The Philosophy of Financial Responsibility" and "A Brief History of Receipts: From Papyrus to PDF."

You've become the educational equivalent of that friend who can't tell a simple story without explaining everyone's

backstory, the political climate, and why the restaurant they went to is historically significant.

Your Enough Bucket isn't just full. It's archaeological. Layer upon layer of "essential context" that only you think is essential.

STEVE'S EPIC FAIL

I once built a "quick refresher" on the new expense reporting system that ended up having prerequisites... that had prerequisites.

The prerequisite chain was eight modules deep. Eight. For expenses. By the time people finished the prerequisites, the expense they wanted to report was from the previous financial year.

INTERNAL DIALOGUE

"They need the foundation first."

"What if they don't know what a risk assessment is?"

"Can't skip steps—that's dangerous."

"Better to over-prepare than under-inform."

"If they don't understand the context, they'll fail."

"Learning must be sequential. It's just logical."

REALITY CHECK

"I just wanted to submit my lunch receipt. Why do I need to understand the philosophical foundations of financial management?"

"This feels like I'm being held hostage by someone's PhD thesis on expense reporting."

"I've been doing expenses for five years. Now I have to prove I

understand what money is before I can use the new button?"

I'd built an educational obstacle course where the prize was learning how to fill out a form. My conditional learning curse had turned a simple task into a master's degree in bureaucratic archaeology.

EARLY WARNING SIGNS

- You use the phrase "before we can cover X" more than "here's how to do X"
- Your modules have prerequisites that have prerequisites
- You design elaborate scaffolding for knowledge people already have
- You feel anxious about people "not being ready" for content
- Your LMS course structure looks like a family tree

RECOVERY ACTIONS

The Trust Test: Assume learners are more capable than your anxiety suggests. Let them try the "advanced" content and see what actually confuses them.

Just-in-Time Learning: Instead of front-loading everything, provide core information upfront and link to deeper context when people need it.

The Assumption Audit: List everything you think learners need to know beforehand. Challenge each assumption with "What evidence do I have that they don't already know this?"

Build Bridges, Not Barriers: Create clear pathways for people who need different levels of support, but don't force everyone through the same foundation course.
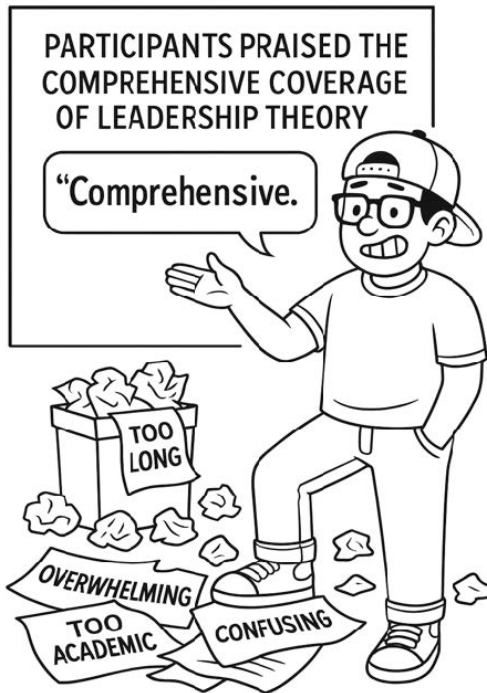
You're not building an escape room. You're building a runway. Let them take off, not trip over your "before you proceed" signs like an over-engineered airport.

11

THE FEEDBACK FIASCO

"People said they liked it!" - Someone with EBD,
defending their 42-tab monument to mediocrity

LET'S SET THE SCENE

You've built something nobody wanted and convinced yourself that one person's polite comment means it's brilliant. Buried in a sea of feedback that screamed "overwhelming," "confusing," and "like being force-fed a management textbook through a fire hose," you found one lukewarm comment: "Comprehensive coverage of leadership theory."

Guess which quote made it into your final report?

"Participants praised the comprehensive coverage of leadership theory, demonstrating the program's effectiveness in delivering thorough leadership education."

You've cherry-picked feedback like you're building a fruit salad of delusion, turning one diplomatic comment into evidence of educational genius.

SYMPTOM DIAGNOSIS

The Feedback Fiasco is classic EBD: convoluted feedback collection plus cherry-picking comments that agree with you. Pure confirmation bias dressed up as evaluation.

You design surveys that make honest criticism socially impossible, then cherry-pick the one diplomatic comment from seventeen brutal ones. "How would you rate the overall learning experience?" forces people to be nice. Nobody wants to destroy someone's hard work in a feedback form.

Meanwhile, you ignore the real feedback buried in the comments: "I just wanted to reset my password, not learn the history of cybersecurity." But that doesn't make it into your stakeholder report.

You've designed a system that protects your ego instead of improving your work. You're not collecting feedback - you're fishing for compliments.

STEVE'S EPIC FAIL

87% hated it. One person said "Comprehensive." Guess which quote made the report.

INTERNAL DIALOGUE

"That one person really got it."

"People just don't appreciate depth."

"Okay, sure, most didn't love it… but that's fine-tuning."

"Comprehensive is good. Confusing just means it's advanced."

"If they didn't enjoy it, maybe they weren't the right learners."

REALITY CHECK

Your survey made it impossible to say what I really thought."

"I wanted to tell you it was too long, but there was no option for that."

"You asked if I 'found value' in the training. What was I supposed to say - no?"

"Your feedback form felt like a trap. Every question assumed I liked it."

"I gave you a 4 out of 5 because 3 felt mean. But honestly? It was a 2."

I'd designed surveys to get the answers I wanted, then used those answers to prove I was right. The one person who wrote "comprehensive coverage" became my entire defence strategy, while seventeen people saying "overwhelming" got ignored.

I'd turned learner politeness into a weapon against honest criticism.

EARLY WARNING SIGNS

- You quote positive feedback more than you analyse negative feedback

- Your evaluation questions focus on satisfaction rather than impact

- You design surveys that make it hard to give honest criticism

- You treat one enthusiastic comment as representative data

- You feel defensive when reading critical feedback

RECOVERY ACTIONS

Level Up Your Questions: Instead of "Did you enjoy this?" ask, "What will you do differently as a result of this training?"

The Reality Check Survey: Ask learners what they'd cut from your module if they had to remove 50%.

The Application Follow-Up: Contact learners 30 days later to ask what they're actually using.

The Honest Broker Test: Find someone who'll give you brutally honest feedback and ask them to review your module.
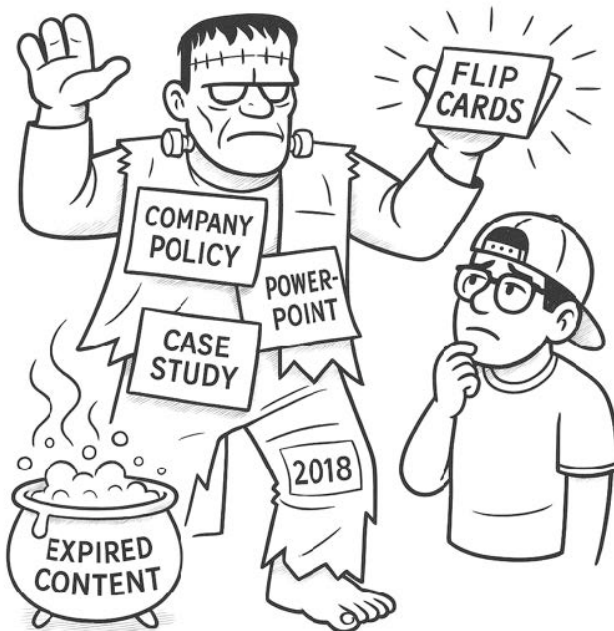
Stop collecting data you're too scared to act on. Stop polishing smiley face metrics. Start asking what matters, and be brave enough to cut the crap, even if someone said they "liked" it.

12

THE FRANKENSTEIN ASSEMBLY METHOD

Expired fish stew, but make it sparkle

SET THE SCENE

"We've already got content for that." "There's a PowerPoint from 2019 that covers most of it." "Let's just copy-paste the policy and jazz it up with some flip cards."

And just like that, welcome to the Reuse Trap - a beautiful, budget-friendly, EBD-fuelled delusion where you take crusty old ingredients and try to pass them off as a gourmet meal.

You're not strategically reusing content. You're panic-patching. You're policy-plastering. You're slapping lip gloss on a zombie and calling it a revitalisation strategy.

SYMPTOM DIAGNOSIS

The Frankenstein Assembly Method is when you raid the content graveyard and stitch together rotting pieces from different eras. A bit of onboarding from 2018, some compliance content from 2020, and that case study everyone loved back when we had offices.

You call it "strategic content reuse." Really, you're Dr. Frankenstein with a PowerPoint licence, except your monster argues with itself across multiple slides.

One section tells people to "always answer the phone within three rings." The next explains your "digital-first communication strategy." You've accidentally built a course that has more personality disorders than a soap opera character.

STEVE'S EPIC FAIL

I created a customer service training monster once. I stitched together five eras of content. It argued with itself halfway through.

INTERNAL DIALOGUE

"This saves time."

"It's all technically still accurate."

"That slide from 2015 still works. Sort of."

"This section was really popular when we used it last."

"No one will notice it's cobbled together."

"It's better than starting from scratch."

REALITY CHECK

"This course feels like it was designed by a committee of people from different decades."

"I started learning about customer service and somehow ended up in a time warp of contradictory advice."

"It's like someone threw a bunch of different courses in a blender and called it learning."

I'd built a content archaeology project that confused learners more than it helped them. My "efficiency" had created educational whiplash.

EARLY WARNING SIGNS

- You use phrases like "we can just repurpose this"
- You justify outdated content with "the principles are still the same"
- You spend more time formatting old content than questioning its relevance
- You build new courses by combining bits of old ones
- You feel proud of your "content efficiency"

RECOVERY ACTIONS

The Content Archaeology Method: Before reusing anything, ask: When was this created? What problem was it solving? Is this still relevant?

The Expiry Date Audit: Put metaphorical expiry dates on all your content. What's past its use-by date?

The Fresh Eyes Test: Give old content to someone who's never seen it. What confuses them?

The Rebuild Challenge: Take your most reused piece of content and rebuild it from scratch. See what you discover.

Reusing content isn't a shortcut. Either do the work to make it genuinely relevant or have the courage to start fresh.

Don't serve reheated sludge and call it training.

13

LAST-MINUTE LARRY'S PANIC ATTACK

"I need it yesterday." Cool. So do I. A holiday.
A metabolism. My dignity.

LET'S SET THE SCENE

t's Thursday afternoon. Larry pops his head over your cubicle wall with that look. You know the one. The "I'm about to ruin your weekend" look.

"Quick question," he says, which is never a quick question. "Can you knock together some training for the leadership workshop? It's on Tuesday."

You stare at him. "This Tuesday? As in, four days from now?"

"Well, three, really. It's just some content and a few slides, right? Nothing fancy. Just... comprehensive. The CEO will be there."

And somehow, against every rational thought in your brain, you hear yourself saying yes because you're a team player. Because everyone's counting on you. Because "something is better than nothing."

Cut to Monday night. You're neck-deep in SCORM exports, surviving on energy drinks and spite, cobbling together slides like you're diffusing a bomb. The final quiz question literally says, "Leadership is good: True or False?" because your brain stopped working six hours ago.

Welcome to Last-Minute Larry's Panic Attack, where impossible deadlines meet your inability to say no, and quality goes to die.

SYMPTOM DIAGNOSIS

Last-Minute Larry's Panic Attack is when impossible deadlines turn you into a caffeinated zombie, building learning experiences with the structural integrity of a house of cards in a hurricane.

Larry doesn't mean to ruin your work. He's just operating from the Crisis Playbook: Everything is urgent. Quality is negotiable. Sleep is optional.

But here's the kicker - you say yes. Every single time. Because the Urgency Illusion has convinced you that any learning is better than no learning. That being fast is more important than being useful. That throwing yourself on the grenade makes you a hero.

You become a content paramedic, performing emergency CPR on dying projects with whatever materials you can grab. The result? Learning experiences that look like they were assembled during a natural disaster by someone having a breakdown.

Which, let's be honest, they were.

STEVE'S EPIC FAIL

Larry once gave me 48 hours to build a "comprehensive leadership development experience." It was 2:48 am. I was naming PNGs after exes and hallucinating navigation buttons. The module launched. So did my anxiety.

I said yes because I'm an idiot.

Did it launch on time? Yes. Did it work? Sort of. Did anyone learn anything? Absolutely not. But Larry got his green checkbox, and I got stress-induced tinnitus.

INTERNAL DIALOGUE

"It's fine. I'll just smash it out."

"There's no time for research, just build."

"Something is better than nothing."

"I'll clean it up later."

"Deadlines matter more than quality."

"I'm the only one who can save this now."

"This is what real heroes do. Mum would be proud."

REALITY CHECK

"This feels like it was built by someone having a nervous breakdown."

"I can tell you rushed this. It shows. In every click."

"Your course doesn't teach leadership. It teaches me that your company doesn't plan ahead."

Larry was happy. The learners were traumatised. I needed therapy.

EARLY WARNING SIGNS

- You agree to timelines that require working through weekends
- You use phrases like "we'll iterate later" knowing you won't
- You skip user research because "there's no time"
- You justify poor design decisions with "it's better than nothing"
- You feel proud of impossible deadline achievements

RECOVERY ACTIONS

The Timeline Reality Check: For your next urgent project, estimate properly, then ask for that time.

Emergency Project Triage: When faced with impossible deadlines, identify the absolute minimum viable outcome and build only that.

The Rush Job Postmortem: Review your last rushed project. What would you do differently with more time?

Scripts for Pushback: "I can deliver something by Friday, but to be clear, it won't be our best work. What's the minimum that would solve the immediate problem?"

Speed isn't the enemy. Stupidity in the name of speed is. You wouldn't build a car by taping wheels to a couch. So why build learning like that?

14

BADGE ME BABY

"Well done, Steve! You clicked Next. Here's a badge.
Go flex that in your next performance review."

SET THE SCENE

You've discovered badges in your LMS. Suddenly, everything needs a reward. Complete Module One and earn the "First Step!" badge for doing what, exactly? Opening a tab?

Module Two? The "Curious Learner!" badge for sitting through an autoplay video while checking your emails?

By Module Six, you've got more badges than a Scout hall, and not one of them means Jack. You've added a confetti cannon that goes off every time someone clicks a flip card. Your course sounds like a New Year's Eve festival goer who's had one too many magic mushrooms.

These aren't motivators. They're participation trophies handed out by insecure designers, hoping no one notices there's no actual game, just a thin layer of glitter glue to a compliance module.

SYMPTOM DIAGNOSIS

Badge Me Baby is a desperate attempt to make people care by throwing shiny things at them. You've discovered the gamification features in your authoring tool and decided everything needs points, badges, and celebratory animations.

But what you get isn't engagement - it's the digital equivalent of giving someone a sticker for breathing. You're not motivating learners. You're patronising them with a very enthusiastic robot.

You wouldn't tolerate this in real life. Imagine your doctor saying, "Congrats, Steve! You turned up to your appointment. Here's a sticker." Or your therapist saying, "You opened up about your trauma? Great job. You've unlocked the Sad Boi badge."

STEVE'S EPIC FAIL

I once built a compliance course with 16 badges. 16 bloody badges. One for logging in. One for breathing. Welcome to gamified mediocrity.

There were badges for:

- "Policy Pioneer" (reading the first policy)
- "Regulation Rockstar" (completing a quiz)
- "Compliance Champion" (watching a video)
- "Safety Superstar" (clicking on a safety tip)
- "Documentation Dynamo" (downloading a PDF)
- And my personal favourite: "Participation Prodigy" (for logging in)

Looking back on this I want to go back in time and punch my alliterative younger self in the face.

INTERNAL DIALOGUE

"Micro-rewards = micro-engagement."

"Everyone loves badges."

"This makes it feel more fun."

"Duolingo does this, and it works."

"I read a blog once that said gamification increases retention."

"A badge for each section keeps them motivated, right?"

REALITY CHECK

"I felt like I was being patronised by a very enthusiastic robot."

"Your badges mean nothing. I got one for clicking a button. That's not an achievement."

"This feels like someone turned mandatory training into a dystopian reward system that celebrates mediocrity."

I'd turned compliance training into a participation trophy factory where everyone was a winner, and nobody learned anything.

EARLY WARNING SIGNS

- Your badges celebrate completion rather than achievement
- You add points without any meaningful way to use them
- Your "leaderboard" ranks people by time spent, not skill gained
- You call something gamification just because it has progress bars
- You feel proud of your "engagement mechanics"

RECOVERY ACTIONS

The Badge Audit: List all your gamification elements and ask, "What real skill or achievement does this represent?"

The Motivation Interview: Ask learners what would actually motivate them to engage with learning.

The Purpose Test: Before adding any game element, ask: What behaviour am I trying to encourage? How does this badge/point/confetti cannon connect to real learning?

The Reality Check: Would you be motivated by this reward or just annoyed?

If you want to gamify, give people a challenge, a reason to play, actual stakes. If you want to engage, give them content that doesn't make them feel like a cog in a compliance machine.
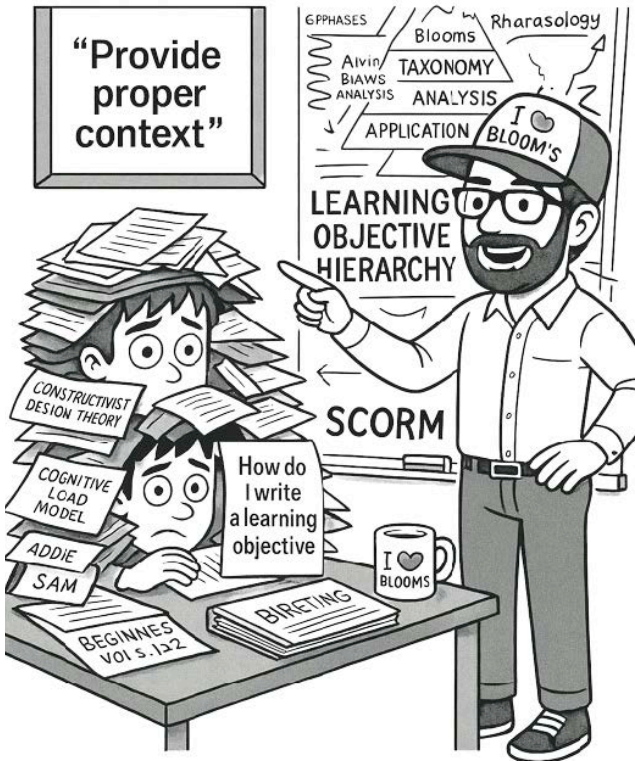
Learners don't care about badges. They care about getting better.

15

THE EXPERT'S PARADOX

When knowing too much makes you teach too much

SET THE SCENE

You've become fluent in the jargon. Comfortable with the complexity. Intimate with the nuances. And now you're designing learning for people who are exactly where you used to be, but you can't remember what that felt like.

You forget that "synergistic stakeholder alignment" isn't English. You assume everyone knows what SCORM stands for. You think a "quick overview of the framework" is actually quick.

Your Enough Bucket isn't just full. It's sophisticated. It's filled with expert-level detail that beginners don't need and can't process. You've built a knowledge fire hose when they needed a water fountain.

SYMPTOM DIAGNOSIS

The Expert's Paradox is advanced EBD. The kind that hides behind expertise and calls itself thoroughness. Once you know something, it becomes nearly impossible to imagine not knowing it.

You've forgotten what it's like to be confused. To feel lost. To desperately need someone to just tell you which bloody button to click without explaining the theoretical framework that underpins button-clicking methodology.

Cognitive scientists call it the Curse of Knowledge. You call it "providing proper context." Your learners call it "educational waterboarding with a side of imposter syndrome."

STEVE'S EPIC FAIL

I once built a "simple introduction" to learning design. It accidentally became a master's degree in instructional theory.

For people who just want to know how to write a learning objective.

INTERNAL DIALOGUE

"They need to understand the theory first."

"This stuff matters. It's foundational."

"You can't write good objectives without Bloom's Taxonomy."

"Okay, maybe this is advanced, but they'll appreciate it."

"Everyone should know this stuff. It's basic.

REALITY CHECK

"I feel dumber now than when I started. I came here to learn the basics and left with a reading list that would take me three years to get through."

"I just wanted to write a learning objective. Why do I need to understand the entire history of instructional design?"

"This feels like you forgot who you were building this for."

I'd built a knowledge avalanche when they needed a stepping stone. My expertise had become their barrier to entry.

EARLY WARNING SIGNS

- You use industry jargon without defining it
- Your "basic" modules require advanced knowledge
- You include "important context" that's only important to experts
- You feel compelled to share everything you know about a topic

- You assume people want the same depth of knowledge you have

RECOVERY ACTIONS

Remember Your First Day: Think back to when you first encountered this subject. What confused you? What did you wish someone had explained differently?

Find Your Beginner: Identify someone who knows nothing about your topic. Watch them try to understand your content. Really watch.

The Jargon Audit: Highlight every technical term in your content. Can you reduce it by 50%?

The Plain English Challenge: Rewrite one section using only common words.

The Expert's Paradox isn't about dumbing down content. It's about smartening up your communication. Your job isn't to prove how much you know. It's to help someone else know what they need to know.

And the mark of true expertise? Making complex things simple, not simple things complex.

CHAPTER

16

THE REAL COST OF EBD

*What we're really doing to the
people we're trying to help*

THE UNCOMFORTABLE TRUTH

W e've been laughing for 14 chapters. And we should. Because if we can't laugh at the shit we've built, we'll never fix it.

But here's the thing we haven't talked about yet. The thing that keeps me awake at 3 am, staring at the ceiling, wondering how many people I've accidentally hurt with my good intentions.

EBD doesn't just waste time. It doesn't just create bad courses. It breaks something deeper. Something harder to fix.

It breaks trust.

WHAT HAPPENS TO LEARNERS

When someone clicks through your Kitchen Sink course and emerges three hours later knowing less than when they started, they don't just forget your content. They learn something else entirely.

They learn that learning doesn't work for them.

They learn that training is something you survive, not something you use.

They learn to keep their head down, tick the boxes, and never volunteer for development again.

And here's the part that really stings: they're not wrong.

THE EROSION OF FAITH

Every bloated module chips away at its faith in learning itself. Every accordion full of irrelevant content makes them a little more cynical. Every forced completion sequence teaches them that their time doesn't matter.

We turn eager learners into learning survivors. People who've been through our content wars come out scarred, suspicious and convinced that professional development is just corporate theatre.

They stop engaging. They stop trying. They start to see learning as something that happens to them, not something they participate in.

THE REAL VICTIMS

It's not just the learners who suffer. It's the next learning designer who has to work with them. The one who builds something genuinely helpful and discovers that their audience has been trained not to trust.

"I don't do training," they say. "It's always just a waste of time."

And they're right because we taught them that.

We built their expectations. We shaped their relationship with learning. We turned development into something you endure rather than something you seek out.

WHAT WE LOSE

When we EBD our way through a project, we don't just lose effectiveness. We lose:

Credibility. Every bloated course makes the next one harder to sell.

Engagement. People learn to switch off, to find ways around learning rather than ways into it.

Impact. Skills that could change careers, improve lives, or solve real problems get buried under layers of unnecessary complexity.

Future opportunities. Learners stop volunteering for development. Managers stop investing in learning. Budgets shrink.

The chance to actually help. Which is why we got into this work in the first place

THE RIPPLE EFFECT

EBD creates a generation of learning-resistant professionals who carry their scepticism into every future development opportunity. They become the managers who say "training doesn't work" and the leaders who cut learning budgets because they've never seen learning actually work.

We don't just waste their time. We steal their potential.

And the worst part? They blame themselves. "I'm just not good at learning," they think. "I must not be the type of person who gets value from training."

We've convinced them that the problem is them, not us.

WHY THIS MATTERS

Every time we choose our ego over their needs, we make it harder for the next designer to build trust. Every Kitchen Sink course creates Kitchen Sink expectations. Every Gavin-infected module teaches learners that training comes with baggage.

We're shaping how people think about learning. And when we get it wrong, we don't just fail. We teach people that learning fails.

WHAT HAPPENS NEXT

Once you see the real cost of EBD, you can't unsee it. Once you understand that every unnecessary accordion is a small betrayal of trust, it becomes impossible to build them.

The learner isn't your user, your audience, or your stakeholder. They're the person whose day you're about to either improve or ruin. Whose faith in learning you're about to either strengthen or destroy. Whose time you're about to either respect or steal.

They deserve better than our anxiety. They deserve better than our ego. They deserve better than our need to look comprehensive.

They deserve stuff that actually works.

And if we can't give them that, we should get out of the way.

The good news? Recognition is the first step toward recovery. And you're almost there.

17

THE EBD FIRST AID KIT

Not a cure, just symptom relief when you're bleeding out

READ THIS FIRST

You're in crisis. Your module has mutated into something that would make Frankenstein weep. It's 11:47 pm, and you're arguing with yourself about accordion animations while your deadline mocks you from tomorrow's calendar.

This isn't a cure. This is a tourniquet. Apply pressure, stop the bleeding, then get professional help.

Pick your symptoms. Apply the treatment. Then, step away from the authoring tool before you make it worse.

FIRST AID PROTOCOL 1: KITCHEN SINK OVERFLOW

Crisis Signs:

- Your *Rise* course has been exported to Storyline "for more functionality"
- You've added 47 different interaction types because "learners need variety"
- Your SCORM package is larger than most people's music collection
- You're using the phrase "immersive learning experience" unironically

Emergency Treatment:

- **Immediate**: Close Storyline. Open a text document. Write the ONE thing learners need to do.
- **Urgent**: Delete every interaction that doesn't directly serve that one thing
- **Critical**: If you can't explain why something exists in one sentence, it goes

Crisis Hotline: "Would I want to deal with this if I was on 3G with a crying baby in the background?"

FIRST AID PROTOCOL 2: GAVIN GREMLIN INFESTATION

Crisis Signs:

- Your "finished" module has grown 200% since sign-off
- Stakeholders are breeding like rabbits in your feedback forms
- You've received 14 "quick thoughts" in the last 2 hours
- Someone just suggested adding a Vyond mascot "for personality"

Emergency Treatment:

- **Immediate**: Create a "Phase Two" parking lot document. Everything goes there.
- **Urgent**: Print your original project brief. Tape it to your monitor.
- **Critical**: For every suggestion, ask: "Does this help achieve the original learning outcome?"

Crisis Hotline: "That's brilliant. Let's save it for Phase Two." (There is no Phase Two. There never was.)

FIRST AID PROTOCOL 3: PERFECTIONISM PARALYSIS

Crisis Signs:

- You're on version 23 of the same slide
- You've spent 6 hours adjusting button hover states in *Captivate*
- You're googling "optimal font size for comprehension" at 2am
- You've exported preview files for yourself to "check"

Emergency Treatment:

- **Immediate**: Set a timer for 30 minutes. When it goes off, you're done.
- **Urgent**: Send the current version to a colleague with subject line "Please publish this"
- **Critical**: Remove your own access to the authoring tool

Crisis Hotline: "Good enough is where learning starts. Perfect is where learning dies."

FIRST AID PROTOCOL 4: CLUTTER OF CARE AVALANCHE

Crisis Signs:

- Your module has more optional content than required content
- You've created safety nets for your safety nets
- There are 23 different resources "just in case someone needs them"
- You feel guilty about deleting a PDF that nobody will read

Emergency Treatment:

- **Immediate**: Count your resources. If there are more than 5, start cutting.
- **Urgent**: Ask: "Does this spark learning or does this spark anxiety?"
- **Critical**: If removing something makes you nervous, that's the stuff to remove

Crisis Hotline: "Love isn't measured by the weight of your content folder."

FIRST AID PROTOCOL 5: PATHWAY PARADOX MAZE

Crisis Signs:

- Your learners need a GPS to find the quiz
- You've built more decision trees than a government department
- People are getting lost in your "user-friendly" navigation
- You need a flowchart to explain your flowchart

Emergency Treatment:

- **Immediate**: Identify the one path that 80% of users should take
- **Urgent**: Delete all other paths. Yes, all of them.
- **Critical**: If you can't navigate it drunk, your learners can't navigate it sober

Crisis Hotline: "Choice paralysis isn't learner empowerment. It's learner torture."

FIRST AID PROTOCOL 6: JUSTIN CASE BUNKER BUILDING

Crisis Signs:

- You're building for the mythical Tasmanian user on dial-up
- Your edge cases have edge cases
- You've created content for problems that don't exist
- Your "just in case" folder is bigger than your actual course

Emergency Treatment:

- **Immediate**: List every piece of conditional content. Delete anything without real evidence of need.

- **Urgent**: Ask: "Has anyone actually asked for this?"
- **Critical**: Build for real users, not imaginary ones

Crisis Hotline: "If someone might need it, they probably won't. And if they do, they'll ask."

FIRST AID PROTOCOL 7: INTERACTIVITY OVERDOSE

Crisis Signs:

- You've added drag-and-drop to content that should be text
- Your confetti cannons are triggered by breathing
- Learners need a degree in interface design to use your interface
- You're describing basic clicks as "micro-interactions"

Emergency Treatment:

- **Immediate**: Remove every interaction. See if the learning still works.
- **Urgent**: Only add back interactions that directly serve learning outcomes
- **Critical**: If it's pretty but pointless, it's poison

Crisis Hotline: "Your JavaScript skills don't need to perform. Your content does."

FIRST AID PROTOCOL 8: EXPERT'S PARADOX KNOWLEDGE BOMB

Crisis Signs:

- You've included the entire theoretical framework for changing a password
- Your "simple introduction" has footnotes
- Beginners are crying into their keyboards

- You assume everyone knows what SCORM stands for

Emergency Treatment:

- **Immediate**: Find a 12-year-old. If they can't understand it, simplify it.

- **Urgent**: Remove every piece of jargon that isn't essential

- **Critical**: Remember what it felt like to be stupid at this stuff

Crisis Hotline: "Your job isn't to prove how smart you are. It's to make them smarter."

EMERGENCY ROOM PROTOCOLS

CODE RED: TOTAL SYSTEM FAILURE

When multiple symptoms hit at once:

1. Close all authoring tools immediately
2. Step away from the computer
3. Touch grass (literally - go outside)
4. Call someone who will tell you the truth
5. Consider starting over with a blank slide

CODE BLUE: STAKEHOLDER CARDIAC ARREST

When Gavin spawns 12 other Gavins:

1. Declare "content freeze" immediately
2. Schedule one feedback session with everyone in the room
3. Bring printed copies of the original brief
4. Practice saying "That's out of scope" in the mirror

CODE BLACK: DESIGNER BREAKDOWN

When you've lost all perspective:

1. Send your current version to three different people

2. Ask them to tell you what to cut

3. Do what they say without arguing

4. Book therapy

THE NUCLEAR OPTION

If all protocols fail and you're still drowning in your own good intentions:

Open a new document. Write this:

"Hi [Learner Name], here's how to [do the thing]. Step 1: [thing]. Step 2: [other thing]. Step 3: [final thing]. Done."

Send it. Job finished. Go have a beer.

Sometimes, the best learning design is no design at all.

EPILOGUE: THE AUTHOR'S RELAPSE

Or: How I Had an EBD Flare-Up Writing About EBD Recovery

THE CONFESSION

Right. So my editor reviewed this book and said, "Steve, it's perfect. Don't touch it."

And what did I do?

I added an epilogue.

Because apparently, I have the self-control of a caffeinated squirrel in a content warehouse. I literally wrote a book about not adding unnecessary shit, then immediately started adding unnecessary shit.

I'm not a hypocrite. I just have EBD.

THE RECOVERY RESOURCES I NEARLY ADDED

I was going to include:

- Quick reference guides (because obviously you need more guides)
- The EBD Emergency Flowchart (a flowchart, for a book that mocks flowcharts)
- Daily prevention checklists (more checkboxes!)
- Severity level classifications (medical-grade content addiction taxonomy)
- Implementation templates (the irony is delicious)

But then I caught myself. So instead, I put them on the Learn Awesome website for you to grab if you want: **www. learnawesome.com.au/ebd**

There. I've successfully avoided Kitchen Sinking my own book about Kitchen Sinking by... creating a separate content warehouse.

I'm basically a reformed hoarder who cleared out their house by renting a storage unit.

APOLOGIES TO THE HUMANS

Dear Gavin, If your name is actually Gavin and you've never super-spread a single stakeholder suggestion, I apologise. Your evil namesakes have ruined it for everyone.

Dear Karen, To all the Karens in Compliance who are actually reasonable human beings - sorry. Your disclaimer-addicted doppelgangers have given you a bad reputation.

Dear Larry, If you're called Larry and you've never panicked about last-minute changes, well done. You've escaped the family curse.

Dear Justin, For the Justins who don't live in constant fear of edge cases - congratulations. You're free from the Case family paranoia.

The names represent archetypes, not people. Though if you recognise yourself AND have the name... that's just cosmic comedy.

APOLOGIES TO THE AUTHORING TOOLS

Dear Every Authoring Tool Company Ever, Stop making your software so bloody irresistible. Every new feature is like giving cocaine to content addicts. We see "New: Advanced Branching Scenarios", and we lose our minds. Your updates are basically EBD triggers disguised as "improved functionality." You know exactly what you're doing.

Stop releasing shiny new features designed to give us EBD flare-ups.

(But don't actually stop. We need the dopamine hits and justifications for our budgets and something to show the CEO so they think we're innovative.)

THE MERCH ANNOUNCEMENT

Speaking of things we didn't need but absolutely had to create - we're launching EBD merchandise! Because nothing says "I understand irony" quite like commercialising minimalism.

THE FINAL CONFESSION

This epilogue is proof that recovery is a journey, not a destination. I'll probably relapse next week and add a companion podcast about the epilogue of the book about not adding things.

If this upsets any of you - people or authoring tools - don't get mad... get merch.

P.S. The website resources are actually quite good, though. Just saying.

P.P.S. I'm doing it again, aren't I?

P.P.P.S. Maybe I should write a sequel about recovering from EBD recovery resources. "The Enough Bucket 2: Electric Boogaloo." No, wait, that's definitely EBD talking. Steve, put the keyboard down. Step away slowly.

ENOUGH'S ENOUGH

*A*send-off and a reminder that clarity is kindness

So here we are.

You've laughed. You've cringed. You've recognised yourself in every epic fail and nodded along to every internal monologue of delusional justification.

The whole book has been the solution. You know what to do. You've always known.

Here's what I've learned after years of building monuments to my own ego:

The best learning design is invisible.

Learners don't notice brilliant navigation, clever interactions or comprehensive coverage. They notice when something works when it respects their time when it helps them do their job better without making them question their will to live.

They notice when you've built something for them, not for your ego.

They notice when you've had the courage to stop.

So build for them. Cut for them. Stop for them.

And when you catch yourself hovering over the "Add Content" button at 11:59 pm, wondering if you should include just one more thing, just one more resource, just one more helpful accordion…

Remember this moment. Remember this field manual.

Take your finger off the mouse. Close the tab. Step away from the keyboard.

You don't need to add anything else.

The hardest part isn't knowing when to stop. It's actually stopping.

That's enough.

ABOUT THE AUTHOR

Steve Corney has designed enough learning to know what too much looks like.

He's been elbow-deep in bloated eLearns, built compliance modules that could double as sleep aids and has hit publish on more than one Frankenstein course stitched together by committee. He's also torn those same beasts apart, rebuilt them with care, and taught other poor souls how to do the same.

A learning designer by trade and a serial over-deliverer in recovery, Steve is the founder of Learn Awesome, a learning agency on a mission to simplify the complex without dumbing it down. He's a TEDx speaker, team wrangler, and unlicensed EBD support group facilitator.

He doesn't have all the answers, but he's got a few hard-won methods that can stop the symptoms of Enough Bucket Dysmorphia from spreading across your next project like glitter in a carpet.

HIS EBD CONFESSION

While writing this book about not adding too much content, Steve suffered a severe relapse during the editing phase. After his editor said, "It's perfect, don't touch it," he immediately added recovery resources, emergency protocols, and this entire section, proving that even EBD experts are not immune to EBD.

He is currently 4 days clean from unnecessary accordions and counting.

WANT TO HANG OUT IRL?

Steve delivers keynotes, conference sessions, and workshops that'll help your team recognise their own EBD symptoms before they infect an entire learning program. Think of it as

group therapy for learning designers but with more laughs and less crying.

Available formats:

- **Keynote presentations** (for when you need to traumatise an entire conference audience at once)

- **Team workshops** (perfect for collective EBD intervention and bonding over shared design trauma)

- **Conference sessions** (because misery loves company, especially when it's educational misery)

- A **Minimalistic eLearn** (with limited accordions and button hover states)

Book an EBD intervention for your team, or just have a stickybeak at what Steve's up to at **stevecorney.com** or **learnawesome.com.au**

Warning: Sessions may cause sudden awareness of your own design habits, uncontrollable urges to delete content, and occasional fits of clarity. Side effects include improved completion rates and happier learners.