The Design of Everyday Things

A 20-Minute Read Summary

By Don Norman — first published 1988 as *The Psychology of Everyday Things*; revised edition 2013



The Book in 60 Seconds

Every day you fight with objects that were designed by clever people who never watched a real person use them. You push a door that needs to be pulled. You stand in a hotel shower turning knobs at random, hoping one of them is hot water. You tap a button on a website, nothing happens, so you tap it again — and submit the form three times.

Don Norman, a cognitive scientist, argues that these are not your failures. They are design failures. When a person cannot figure out how to use something, the object is communicating badly. Good design tells the truth about how it works. Bad design lies, and then blames you for believing it.

This book gives you the vocabulary to name what is wrong — affordances, signifiers, mapping, feedback, conceptual models, the gulfs of execution and evaluation — and a humane principle to judge everything by: human error is usually design error. The designer's job is to make the right action easy, the wrong action hard, and recovery from mistakes trivial. First published in 1988 and revised in 2013, it helped create the field of user-experience design, and it remains the book people hand you when you ask why so much of the designed world feels hostile.

The Story: A Door That Defeats You

Norman opens with doors, because a door has one job — open, close, let people through — and we have somehow managed to make doors confusing. The worst offenders are what he calls Norman doors: a glass door with a handle that invites you to pull, but which only opens if you push. A door with no handle at all, so you stand there patting the glass. A door with a handle on both sides, one of which does nothing.

The term is named after Norman himself, and it is self-deprecating on purpose. He spent years walking into such doors. His point is that the door is communicating, whether the designer meant it to or not. A handle says "pull." A flat plate says "push." When the signal and the mechanism disagree, the user loses every time.

The door is not an exception. It is the rule. Norman's argument is that every object you touch is telling you something about how it works, and most objects tell you the wrong thing — or nothing at all. Design is communication. A good design tells the truth. A bad design lies to you and then blames you for believing it.

The Core Ideas

Affordances: What a Thing Lets You Do

An affordance is what an object lets you do with it. A chair affords sitting. A flat surface affords pushing. A handle affords pulling. A knob affords turning. The idea comes from the psychologist James Gibson, who studied how animals perceive their environment.

Norman's crucial addition is that affordances can be real or perceived — and for design, perception is what matters. A door can physically be pushed, but if it looks like it should be pulled, the design has failed, even though the door technically works. The designer has to work with what people perceive, not just what is physically true.

Signifiers: Signals That Say Where to Act

If affordances are what an object allows, signifiers are the signals that tell you where to act and how. The word "push" written on a door is a signifier. So is the shape of a handle, a highlighted button on a screen, or the raised dots on a keyboard that mark the home keys.

Good signifiers are obvious and honest. Bad signifiers are missing, misleading, or so subtle you need a manual to find them. Once you start noticing signifiers, you see how often they are absent. A hotel shower with three unlabeled knobs and a lever asks you to experiment on your own body with no instructions. That is a design failure dressed up as a luxury experience.

Mapping: Controls Should Match What They Control

Mapping is the relationship between controls and their effects. A stove top has four burners in a square. If the four knobs are arranged in a matching square, you know instantly which knob controls which burner. If the knobs are in a straight line, you have to guess — and you will guess wrong about half the time.

Good mapping is natural and spatial. Bad mapping forces you to memorize an arbitrary code. The stakes change with the context — an annoyance on a stove, a danger on a hospital ventilator or an airplane cockpit — but the psychology is the same.

Feedback: Tell Me It Worked

Feedback is the information a system gives you about the result of your action. A button that clicks when you press it gives feedback. A light that turns on gives feedback. A door that swings open gives feedback.

The absence of feedback is one of the most common design failures in the world, and it is especially common in digital products. You tap a button on a website, nothing visibly happens, so you tap it again — and submit the same form three times. Feedback should be immediate, visible, and proportional to the action.

Conceptual Models: The Picture in Your Head

You carry a mental picture of how things work — a conceptual model. Your model of a thermostat is probably that turning it up higher makes the room heat up faster. That model is wrong. A thermostat is an on-off switch with a target temperature; turning it to ninety does not heat the room faster than turning it to seventy-two. It just overshoots.

The designer built the system with one model in mind; you arrived with a different one. The gap between the designer's model and the user's model is where frustration lives. The designer knows how the system works, so everything seems obvious to them. The user sees only the surface — what Norman calls the system image. If the system image does not communicate the right model, the user will invent a wrong one. People do not invent wrong models because they are stupid. They invent them because the design gave them nothing better to work with.

Visibility: The Master Principle

If the book has one master principle, it is visibility. The right things must be visible: what actions are possible, what the current state is, and what will happen if you act. Norman calls this discoverability, and it is really the summary of the whole book. Can a person look at this thing and discover what it does, how to do it, and whether it worked? If the answer is yes, the design is good. If the answer is no, no amount of clever engineering underneath will save it.

Visibility explains why the modern world is so often frustrating. Functions have disappeared behind flat glass, hidden menus, and gestures with no visible trace. The hamburger menu — those three little lines in the corner of an app — hides everything behind a tap. A swipe left does one thing, a swipe right does another, a long press does a third, and nothing on screen tells you any of it. A control you cannot see is a control that does not exist for the user.

Norman's point is not that screens are bad. It is that designers removed the visible cues that physical objects used to provide — the shape of a button, the position of a dial, the label next to a switch — and replaced them with nothing. Clean is not the same as clear. Minimal is not the same as understandable. A beautiful interface that hides its functions is a locked door with no handle: elegant, and useless.

The fix is not complicated, and Norman repeats it in different forms throughout the book: make the possible actions visible, make the current state visible, and make the result of each action visible. When those three are true, people rarely get lost. When any one of them is missing, people get lost constantly — and blame themselves for it.

The Seven Stages of Action — and the Two Gulfs

Whenever you use a device, Norman says, you pass through seven stages: you form a goal, form an intention, specify an action, execute it, perceive the state of the world, interpret what you perceived, and evaluate whether it matches your goal.

The model sounds academic, but it is a practical diagnostic tool. The failure almost always happens at a specific stage. You could not figure out what to do — a failure of execution. Or you did something and could not tell whether it worked — a failure of evaluation. Norman calls the gap on the execution side the gulf of execution and the gap on the evaluation side the gulf of evaluation. Good design bridges both gulfs: the control is visible and its function is clear, and when you act, the result is immediately obvious.

Imagine a rental car at night. You want the headlights on. The controls are hidden on a stalk behind the wheel, marked with tiny symbols you cannot read — a wide gulf of execution. You find a dial and turn it. Nothing obviously changes — a wide gulf of evaluation. A good design narrows both. The control is visible and its function is clear, and when you act, the result is immediately obvious.

A worked example makes the model concrete. Picture an office photocopier — the classic Norman villain. Your goal is to make ten double-sided copies of a report. Your intention is clear. But the copier's control panel has forty buttons, most of them labeled with abbreviations, and the double-sided option is buried three menus deep. That is the gulf of execution: you cannot figure out how to turn your goal into action. Suppose you find the option and press start. The machine whirs, but the display shows a cryptic code and no count of pages completed. You stand there wondering whether it is working, whether it jammed, whether it is copying the right pages. That is the gulf of evaluation: you cannot tell what state the system is in or whether your goal was achieved. Every stage of the seven is a place where the design can fail you, and on a bad copier, it fails you at most of them.

The model also explains why good design feels effortless. When the gulfs are narrow, you move from goal to evaluation without thinking — the action is obvious, the feedback is immediate, and the whole interaction disappears into the background. That invisibility is the reward of good design, and it is also why good design is so hard to notice. You only see design when it fails.

Slips, Mistakes, and Forcing Functions

Norman's most humane idea is that human error is usually design error. He divides errors into two kinds. A slip is when you have the right intention but the wrong execution — you meant to press save and pressed delete because the buttons sit next to each other. A mistake is when the intention itself is wrong, because your model of the system was wrong — you turned the thermostat to ninety because you believed it would heat the room faster.

The designer's job is to handle both. For slips, make the dangerous action harder and the safe action easier: put delete far from save, or ask for confirmation. For mistakes, fix the conceptual model: redesign the thermostat so its behavior is visible. The general rule is simple — make errors hard to commit and easy to recover from.

The strongest version of this is the forcing function: a constraint that physically prevents an error. Your car will not start unless it is in park. Your microwave will not run with the door open. Forcing functions do not rely on the user's attention or memory; they make the error impossible at the level of physics. Norman argues designers should use them wherever the cost of an error is high — and that too many systems rely on warnings and labels instead, which people ignore because they have seen a thousand warnings and most of them were meaningless. A warning is an admission of design failure. If you have to warn people not to do something, you have already built a system that invites them to do it.

Constraints: Four Ways to Guide the User

Good design uses constraints to guide you toward the right action, and Norman names four kinds. Physical constraints limit what is possible: a puzzle piece only fits in one place. Semantic constraints rely on meaning: red means stop. Cultural constraints rely on shared conventions: in English we read left to right, so a sequence of controls reads left to right. Logical constraints rely on reasoning: four burners and four knobs arranged to match.

The best designs use all four at once, so the correct action feels natural and the wrong action feels wrong. A pair of scissors can barely be held the wrong way. A modern phone charger plug is symmetrical, so there is no wrong way to insert it. That is a design that removed the error entirely instead of asking you to be more careful.

Knowledge in the World vs Knowledge in the Head

People cannot remember arbitrary procedures. People get interrupted, tired, distracted, in a hurry. A design that assumes a calm, attentive, fully informed user is a design for a person who does not exist.

Good design puts the knowledge in the world, where you can see it, instead of demanding that you carry it in your head. A well-designed door puts the knowledge in the world: the handle tells you to pull. A badly designed door demands knowledge in the head: you must remember that this particular door, unlike every other door you have met, opens by pushing. The more a design forces you to memorize, the worse it is.

Great design is discoverable. You can look at it and figure out what it does, without a manual, without a tutorial, without having used it before. Norman suggests a test: watch someone use a thing for the first time without helping them. If they struggle, the design is at fault. The first-time user is the truth teller — and the person the design is actually for.

The Paradox of Technology

As technology gets more powerful, it offers more functions, and more functions mean more complexity, and more complexity means a harder design problem. The early remote control had a few buttons; the modern one has dozens. The early car had a few controls; the modern car has menus inside menus. Each new feature is genuinely useful to someone, but the total experience gets harder. The designer's job is to manage that paradox — to hide complexity without removing power, and to make the common actions easy even if the rare actions are hard.

The revised edition adds the modern version of this problem: the tyranny of the touch screen. A physical button has a shape you can feel without looking. A touch screen is a flat piece of glass; every function is hidden, and the only way to discover what you can do is to tap around and hope. Carmakers moved climate controls onto touch screens, partly because it looks futuristic and partly because it is cheaper — but a driver cannot feel a touch screen. To change the temperature you must take your eyes off the road. The screen is more powerful and less usable.

The Case Study: Three Mile Island

The stakes of all this are not theoretical. In 1979, the nuclear power plant at Three Mile Island in Pennsylvania suffered a partial meltdown. The control room had hundreds of identical-looking switches and dials, with no clear mapping, poor feedback, and alarms that all sounded the same. When the emergency happened, the operators were drowning in information and starved of understanding. They misread the situation — not because they were incompetent, but because the interface made the truth nearly impossible to see.

The official report blamed the operators. Norman says the report was wrong. The design was the failure. The same psychology that makes a door confusing makes a control room dangerous. Good design is not a luxury or a matter of taste. In high-stakes systems it is a safety issue. In everyday life it is a dignity issue, because every time a design makes you feel stupid, it takes something from you.

Human-Centered Design

Norman's answer is a process, not a single insight. Human-centered design starts by observing real people using real things in real situations. You build prototypes. You test them with users. You watch where they stumble, and you redesign. Then you test again.

The key is that you never trust your own intuition about what is obvious, because to the designer, everything is obvious. The designer knows how it works. The test is whether a stranger can use it. Human-centered design is humble design: it assumes you will be wrong, and it builds in a way to find out.

The revised edition extends this to complex systems. A hospital is not one device; it is dozens of devices, hundreds of people, and software systems that do not talk to each other. A single well-designed device can still fail inside a badly designed system. That is a bigger problem than doors, and Norman admits it is harder — but the principles are the same: observe real people, find out where they stumble, redesign.

Why It Matters

This book landed on lists of the best product management books for a reason. If you build products, or manage people who build products, it is the foundation. It is why we talk about user experience as a discipline, why companies hire designers and researchers instead of treating design as decoration. Norman himself worked at Apple in the 1990s as a vice president in the advanced technology group, and later co-founded the Nielsen Norman Group with Jakob Nielsen, still one of the most respected design research firms in the world. User experience — as a job title, as a budget line — owes a debt to this book. Before Norman, design was mostly about how things looked. After Norman, design had to be about how things worked for the person using them.

For a product manager, the book is a checklist to carry into any product review. Is it discoverable? Does it communicate its conceptual model? Is the mapping natural? Is the feedback immediate? Are errors hard to make and easy to undo? If the answer to any of those is no, you have work to do, and no amount of clever marketing will cover it.

And if you are not a designer at all, the book matters even more simply. You live inside designed things all day — your home, your car, your phone, your kitchen, your office. When those things are well designed, your day is smoother and you barely notice. When they are badly designed, you pay for it in small frustrations that add up. This book gives you the language to name what is wrong, and the confidence to stop blaming yourself. You were not stupid. The door was.

There is a final, quieter gift in the book, and it is the way it changes your relationship with frustration itself. Before reading Norman, a confusing microwave or an unopenable door feels like a personal failing — a small, private embarrassment you hurry past. After reading him, the same moment becomes evidence: a data point about a design that failed its user. That shift, from shame to diagnosis, is surprisingly freeing. It turns hundreds of tiny daily irritations into something you can observe, name, and even enjoy spotting, the way a birdwatcher enjoys spotting a rare species. The world does not get less badly designed. You just stop taking it personally.

How to Approach Reading It

Get the revised edition from 2013, which is the one in print now. The first three chapters are the core — that is where Norman lays out affordances, signifiers, mapping, feedback, conceptual models, and the seven stages of action. Read those slowly. The middle of the book is examples, and you can dip in and out of them. The later chapters cover the design process, human-centered design, and the paradox of technology. If you are in a hurry, read the first three chapters and the final chapter, and you will have most of the value.

Read it with your eyes open, literally. As you go through your day, notice the doors, the stoves, the showers, the apps. Notice where you hesitate, where you guess, where you have to think. Every one of those moments is a design failure you can now name. This is one of those rare books that changes what you see, and once you see it, you cannot unsee it.

Memorable Lines

Norman's most quoted lines are really design principles stated as plainly as possible:

- "A brilliant solution to the wrong problem can be worse than no solution at all."
- "Good design is actually a lot harder to notice than poor design, in part because good designs fit our needs so well that the design is invisible."
- "The design of everyday things is in great danger of becoming the design of superfluous, overloaded, unnecessary things."
- On the famous teapot with its handle beside its spout, designed by Jacques Carelman: a "teapot for masochists."
- "Rule of thumb: if you think something is clever, think again."
- "Human-centered design is a philosophy, not a set of steps."

Fair Criticisms

It is a way of seeing, not a how-to manual. The book tells you how to think about design, but it will not teach you to run a usability study in detail, prototype an interface, or manage a design team. You will finish it and see the world differently, but you will still need other books and a lot of practice to become a designer.

Usability is not the whole story. One fair criticism is that the book can make design sound purely functional, as if the only question is whether the thing works. Norman himself came to think that was incomplete, and his later book *Emotional Design* argues that beauty, pleasure, and the feeling of a well-made object matter too. A great design is usable *and* delightful.

The affordance debate. Norman borrowed "affordance" from the psychologist James Gibson, who meant something slightly different — what the environment actually offers an animal, not what a person perceives. Scholars still argue about the distinction. For a general reader it matters less than the practical idea: design for what people perceive, not just what is physically true.

Some examples have aged. The telephones, washing machines, and video recorders of the original edition are museum pieces now. The revised edition updates many of them, but technology moves faster than books do. What does not change is the psychology: people still cannot remember arbitrary procedures, still get confused when the mapping is wrong, and still blame themselves when the design fails.

Why It Is on the List

The Design of Everyday Things appears on best product management book lists because it is the book that made product thinking human. It established the principle that a product's quality is measured at the point of use, by a real person, in a real situation — not in the engineering lab. It gave product people a shared vocabulary for arguing about usability, and it made the case that watching users is not optional. Nearly forty years after it was first published, it is still assigned in design courses around the world, still in print, and still the fastest way to learn to see the designed world clearly.

What to Read Next

- **Don Norman, *Emotional Design* (2004)** — Norman's own sequel, adding beauty and pleasure to the story: why we love some things and hate others, and why that love is not irrational.
- **Steve Krug, *Don't Make Me Think* (2000; revised 2014)** — the same ideas applied to websites and apps: short, funny, and practical.
- **James J. Gibson, *The Ecological Approach to Visual Perception* (1979)** — the deep psychology behind affordances, for readers who want the theory.
- **Jared Spool and the Nielsen Norman Group's research** — the modern, evidence-based continuation of Norman's program: what usability studies actually find.