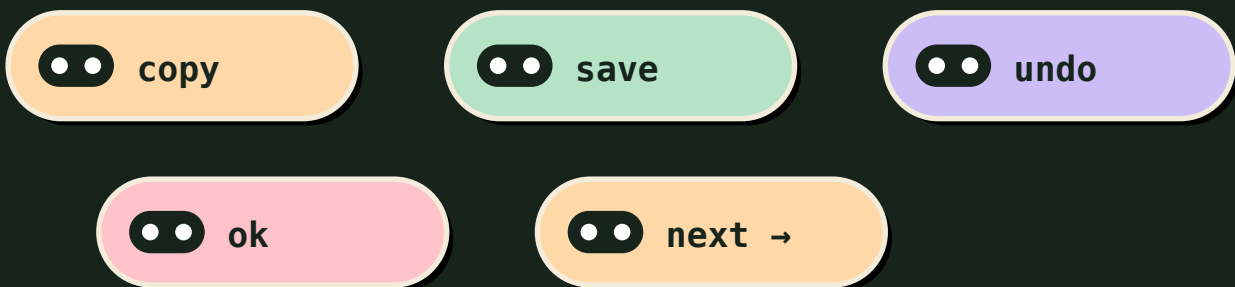


Micro- interactions That Matter

Thirty button patterns, their anatomy, and how to test them so they work for everyone who presses them.



HOW TO USE THIS GUIDE

A field guide, not a rulebook

Buttons are the smallest promises an interface makes. “Save” promises the thing will be there tomorrow. “Delete” promises it won’t. Every press is a tiny contract, and the microinteraction — the half-second of motion, sound, label change or silence that follows — is how the interface shows it kept its word.

This guide accompanies **Button Zoo** (buttonzoo.sahiltayal.online), an exhibition of thirty original button behaviours that each perform a real action. It is written for designers, developers, product people and curious students who want to make controls that are expressive *and* dependable. Every pattern below exists as working code on the site; copy the snippets there.

What’s inside

Pages	Section
3	Anatomy of a microinteraction
4	Seven principles from the zoo
5-9	The thirty patterns, with “do” and “watch out” notes
10	Reduced-motion translation table
11	Three worked redesigns (do / don’t)
12	The twelve-point accessible button testing checklist
13	Exercises and 24 design prompts
14	References, limitations and colophon

How to read the claims in this guide

Standards (WCAG 2.2, the ARIA Authoring Practices) are cited by success-criterion number so you can check the source text yourself. **Interpretations** — such as which patterns “need care” — are the author’s editorial judgement, built from practice rather than from a study. **Metaphors** (enclosures, specimens, Latin names) are play; they carry no technical meaning. Nothing here is a legal opinion or a certification of compliance.

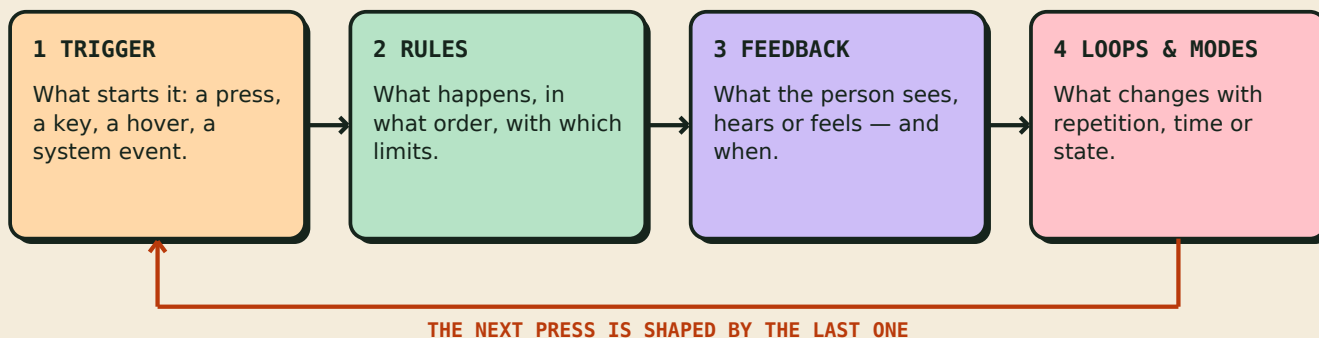
Three habits make the guide most useful: (1) open the matching specimen on the site while you read; (2) turn on your operating system’s “reduce motion” setting at least once and press everything again; (3) try every pattern with only a keyboard before you trust it.

PART ONE

Anatomy of a microinteraction

Dan Saffer's *Microinteractions* (O'Reilly, 2013) gives a durable vocabulary: a microinteraction has a **trigger**, **rules**, **feedback**, and **loops and modes**. Button Zoo prints those four parts on every placard and adds three fields that the original framework leaves implicit but practice demands.

SAFFER'S FOUR PARTS + THE ZOO'S THREE PLACARD FIELDS: KEYBOARD · REDUCED · ACCESSIBILITY RATING



The zoo's three additions

Field	Question it answers	Example from the zoo
Keyboard	What does someone without a pointer experience — and is it equivalent?	#14 Hold-to-export: hold Space/Enter, or press twice within 4 s.
Reduced	With “reduce motion” on, what replaces the movement so the meaning survives?	#11 Ripple: no expanding circle; the prompt text cross-fades.
Accessibility rating	How easily does this pattern go wrong in other hands? (low / watch closely / needs care)	#27 Shy button: needs care — fleeing targets are hostile unless fenced.

A worked anatomy: the Two-Step Heron (#18)

Trigger	Press “Clear notebook”.
Rules	First press arms for 4 s; second press within the window clears; blur or timeout disarms; an Undo toast offers 8 s of recovery.
Feedback	Label becomes “Really clear? Press again”; colour shifts to warning; a polite live region announces the armed state; the toast shows a draining bar and a Ctrl/⌘+Z hint.
Loops & modes	Armed is a mode. Modes need an exit (timeout, blur) and must be announced, or people will be surprised by them.

PART TWO

Seven principles from the zoo

These are the editorial rules the thirty specimens were built against. They are distilled from accessibility standards and common practice; where a principle rests on a specific standard, the related WCAG criterion is noted on page 14.

01 Feedback within a tenth of a second.

Something visible should change within roughly 100 ms of a press, even if the real work takes longer. A pressed state, a label change, a spinner with words — anything that says “I heard you.”

02 The label names the outcome, not the gesture.

“Copy link to this view” beats “Share”. When a button acts on a target, name the target (“Mark #19 reviewed”), because screen-reader users often hear the button out of context.

03 State is visible without colour alone.

Toggle buttons expose aria-pressed and also change text, shape or icon. The zoo’s Shapeshifter (#9) changes a circle into a crescent; the Tortoise (#23) flips its shell.

04 Motion is a sentence; reduced motion rephrases it.

Ask what each movement communicates — “this settled”, “here is where you touched”, “this came from there” — and carry that meaning with colour, outline or text when movement is reduced.

05 Prefer undo to “Are you sure?”

Confirmation dialogs interrupt everyone to protect a few. Act immediately and offer generous undo where the system allows it; reserve confirmations for what genuinely cannot be reversed.

06 Sound and vibration are invited, never imposed.

Off by default, enabled by an explicit control, never the only carrier of information. Create the `AudioContext` only after the press that permits it.

07 Mischief needs fences.

Play is welcome when it is brief, optional, reversible and never blocks the task. The Shy Lemur (#27) dodges a mouse twice, never a keyboard or a finger — and still works.

“A button is a promise small enough to keep every time.”

PART THREE

The thirty patterns

Each entry gives what the specimen does, the rule worth copying (**Do**), and the way it most often fails (**Watch out**). Ratings: **low risk**, **watch closely**, **needs care**. Code for every pattern is on the site — press “Copy snippet” on its card.

■ ZONE I · MAGNETIC HOUSE — Buttons that lean toward attention without grabbing it.

M01 The Magnet Copier <i>Magnetica copians</i> [watch closely · Pointer events] Copies its own implementation snippet to your clipboard. Within its enclosure it drifts up to 6px toward a mouse pointer. Do — Pull only with a fine pointer; cap displacement at 6px; snap home on leave. Fully operable; magnetism never applies to keyboard focus, so focus rings stay put. Watch out — Moving targets can be harder for tremor or low-precision users. Keep the pull tiny and keep the hit area static. Reduced motion: No drift. The button simply darkens on hover. Label: “Copy my snippet”	
M02 The Lodestone <i>Lodestonia proxima</i> [low risk · Pointer events] Scrolls to the next zone of the zoo and moves keyboard focus to that zone’s heading. Do — Cycle through zones; wrap at the end; focus the heading with <code>tabindex="-1"</code> . Enter/Space; focus lands on the destination heading. Watch out — Scroll-only navigation strands keyboard users. Always move focus too. Reduced motion: Jumps instantly instead of gliding. Label: “Next enclosure →”	
M03 The Iron Filings <i>Ferrum collectivum</i> [low risk · CSS transitions] Saves every specimen currently visible (after your filters) into your Field Notebook in one go. Do — Scope = current filter result; skip duplicates; report how many were new. Enter/Space. The filing animation also plays on <code>:focus-visible</code> . Watch out — Bulk actions need an explicit count in the accessible name, not just in a tooltip. Reduced motion: Filings stay still; only the count updates. Label: “Save all visible”	
■ ZONE II · SPRING PADDOCK — Mass, stiffness, damping: motion that says “this is physical, and it settled.”	
M04 The Pogo Toggle <i>Saltator conspectus</i> [low risk · Web Animations API] Switches the whole gallery between grid and list layouts. The label always names the current view. Do — Two states; persist choice; announce change. Enter/Space; aria-pressed reflects list view. Watch out — Label text and aria-pressed must not contradict each other. Pick one model and stick to it. Reduced motion: No bounce; the layout swaps directly. Label: “View: grid”	
M05 The Slinky <i>Spira densitatis</i> [low risk · CSS transitions] Cycles gallery spacing through compact → cozy → airy. Do — Cycle 3 values; label shows current value. Enter/Space. Watch out — A cycling button hides the other options. Fine for 3; use a radio group beyond that. Reduced motion: Spacing changes without the coil animation. Label: “Density: cozy”	
M06 The Jack-in-the-Box <i>Pyxis exsiliens</i> [low risk · Web Animations API] Opens the full field-guide entry for today’s specimen (chosen deterministically from the date). Do — Same specimen all day; modal <code><dialog></code> ; focus returns on close. Enter/Space opens; Esc closes; focus returns to this button. Watch out — Modals must trap focus and restore it. Native <code><dialog>.showModal()</code> handles most of this. Reduced motion: Dialog fades in without scale. Label: “Specimen of the day”	

M07 The Bungee *Funis reditus* [low risk · Web Animations API]

Returns to the top of the page and focuses the main heading.

Do — Scroll + focus <h1>. Enter/Space; focus moves to the page title.

Watch out — Don't make it a floating overlay that covers content on small screens.

Reduced motion: Instant jump, no stretch.

Label: "Back to the gate ↑"

■ **ZONE III · MORPHING GLASSHOUSE** — *One control, several honest shapes. The shape is the state.*

M08 The Chrysalis *Pupa tokenorum* [low risk · CSS transitions]

Copies the zoo's CSS motion tokens (durations, easings, reduced-motion overrides) as design code.

Do — Morph label for 1.6s then revert; never shift surrounding layout. Enter/Space; result announced via live region.

Watch out — Label changes are not always announced; pair them with a polite live region.

Reduced motion: Label swaps without width animation.

Label: "Copy motion tokens"

M09 The Shapeshifter *Lux tenebraeque* [low risk · CSS transitions]

Toggles light and dark themes. Its icon morphs between a full circle (sun) and a bitten crescent (moon).

Do — Persist; respect OS preference until the user chooses. Enter/Space; aria-pressed = dark.

Watch out — Both themes must independently pass contrast checks, including focus rings.

Reduced motion: Icon swaps without sliding.

Label: "Theme: light"

M10 The Unfolding Pill *Ordo explicans* [watch closely · CSS transitions]

Unfolds into three sort choices (number, zone, risk). Picking one re-sorts the gallery and folds back.

Do — aria-expanded on trigger; options are aria-pressed buttons; Esc folds. Enter opens, Tab through options, Esc closes and returns focus.

Watch out — Disclosures must expose aria-expanded and must not hide focus inside collapsed content.

Reduced motion: Options appear without width animation.

Label: "Sort: number"

■ **ZONE IV · RIPPLE POND** — *Feedback that radiates from exactly where you touched.*

M11 The Pond Skipper *Unda inventionis* [low risk · CSS transitions]

Draws a new button-design challenge from a deck of 24 original prompts and shows it on the Prompt Board.

Do — Never repeat the previous prompt; keyboard ripple starts at centre. Enter/Space ripple from the centre.

Watch out — The ripple is decoration; the new prompt text is the real feedback and must be announced.

Reduced motion: No ripple; the board text cross-fades.

Label: "Shuffle a design prompt"

M12 The Sonar *Echo inveniens* [low risk · CSS transitions]

Highlights every specimen already in your notebook, scrolls to the first, and announces how many it found.

Do — Pulse lasts 2.4s; announce count, including zero. Enter/Space; focus moves to the first matching card.

Watch out — Highlight with outline thickness too, not colour alone.

Reduced motion: Static thick outline instead of pulsing rings.

Label: "Ping my saved specimens"

M13 The Echo *Resonantia communis* [low risk · CSS transitions]

Copies a URL that reproduces your current filters and sort (stored in the #hash). Paste it anywhere to share the exact view.

Do — Encode zone/purpose/technique/risk/sort; no personal data in the URL. Enter/Space.

Watch out — If the Clipboard API is blocked, fall back to showing a selectable text field.

Reduced motion: No rings; label change only.

Label: "Copy link to this view"

■ ZONE V · TACTILE BURROW — *Pressure, holding, buzzing and the size of a fingertip.*N14 The Patient Burrower *Tenax exportans* [needs care · Timers / state machine]

Press and hold (0.9s) to download your Field Notebook as a Markdown file. Prefer not to hold? Press twice within 4 seconds.

Do — Cancel on release before 0.9s; second short press within 4s also exports. Hold Space or Enter, or press twice.

Watch out — WCAG 2.5.1 / 2.5.7 spirit: never require a timed or path gesture without a simple alternative.

Reduced motion: Fill bar steps instead of sweeping.

Label: “Hold to export notebook”

N15 The Rubber Stamp *Sigillum promptorum* [low risk · CSS transitions]

Saves the design prompt currently on the Prompt Board into your notebook (with today’s date).

Do — Skip if that prompt is already stamped. Enter/Space; :active styling also triggers via a .pressed class on keydown.

Watch out — :active rarely fires for keyboard in all browsers, so mirror it with a class.

Reduced motion: Colour flash instead of squash.

Label: “Stamp prompt into notebook”

N16 The Buzzing Beetle *Vibratio permissa* [watch closely · Vibration API]

Turns on optional 15ms vibration feedback for zoo actions — only where the browser supports it (mainly Android). It tells you honestly if your device can’t.

Do — Off by default; feature-detect; never vibrate without a user gesture. Enter/Space; aria-pressed.

Watch out — Never use vibration as the only feedback. iOS Safari does not support navigator.vibrate.

Reduced motion: Unchanged (vibration isn’t motion on screen), but you can leave it off.

Label: “Haptics: off”

N17 The Fingertip Gauge *Mensura digiti* [low risk · DOM reorder]

Overlays every button and link on this page with a size verdict: green ≥ 44 px, amber 24–43px, red under 24px. Reports totals.

Do — Measure getBoundingClientRect; inline text links are flagged but may qualify for the inline exception. Enter/Space; aria-pressed.

Watch out — A measurement aid, not a compliance certificate: spacing exceptions need human judgement.

Reduced motion: Unchanged (static overlay).

Label: “Inspect touch targets”

■ ZONE VI · CONFIRMATION AVIARY — *The pause before an action that cannot simply be taken back.*N18 The Two-Step Heron *Ardea bis-tangens* [watch closely · Timers / state machine]

Empties your Field Notebook. First press arms it (“Really clear? 4s”); second press clears. An Undo is offered afterwards.

Do — Auto-disarm after 4s or on blur; announce armed state. Enter/Space twice. Focus never moves between steps.

Watch out — Announce the armed state; a silent colour change is invisible to screen-reader users.

Reduced motion: No shake; colour and text change only.

Label: “Clear notebook”

N19 The Inkwell Checkmark *Signum perlectum* [low risk · SVG stroke]

Marks the specimen you most recently opened as “reviewed” (persists in this browser). Press again to un-mark.

Do — Target = last opened specimen; label names the target. Enter/Space; aria-pressed reflects reviewed state of the target.

Watch out — Name the target in the label. “Mark reviewed” alone is ambiguous out of context.

Reduced motion: Checkmark appears fully drawn.

Label: “Mark #19 reviewed”

N20 The Weighing Stork *Pondus documenti* [low risk · Timers / state machine]

First press tells you exactly what you’ll get (page count, file size, format). Second press downloads the PDF.

Do — Disarm after 6s; a plain direct link also exists for everyone. Enter/Space twice.

Watch out — Always keep a plain, single-step link to the same file elsewhere on the page.

Reduced motion: No expansion animation.

Label: “Get the field guide (PDF)”

■ ZONE VII · REVERSIBLE REEF — *Act now, regret safely. Undo beats “Are you sure?”*M21 The Hermit Crab *Paguroidea revertens* [low risk · Timers / state machine]

Removes the most recent notebook entry immediately, then offers an Undo toast for 8 seconds.

Do — Act instantly; keep removed item in memory 8s; Undo restores to the same position. Enter/Space. Toast Undo is reachable by Tab and by pressing Ctrl/⌘+Z.

Watch out — Toasts that vanish can fail WCAG 2.2.1 (timing). Give generous time and a keyboard shortcut.

Reduced motion: Toast appears without sliding; countdown bar is static text.

Label: “Remove last saved item”

M22 The Rewinding Nautilus *Spira temporis* [low risk · Timers / state machine]

Undoes the most recent change to any zoo setting (theme, view, density, motion, sound, mirror...). Keeps a 20-step history.

Do — History stack of {key, previous}; disabled-but-focusable when empty. Enter/Space. When empty it uses aria-disabled, not disabled, so it stays discoverable.

Watch out — Prefer aria-disabled with an explanation over removing a control from tab order.

Reduced motion: No spiral rotation.

Label: “Rewind last setting”

M23 The Tortoise Twin *Testudo comparativa* [low risk · CSS transitions]

Switches the entire zoo between full motion and reduced motion, so you can compare every specimen live. Press again to switch back.

Do — Starts from your OS setting; override stored per browser. Enter/Space; aria-pressed = reduced.

Watch out — An in-page override should never be able to force motion on someone whose OS asks for less without their explicit choice.

Reduced motion: This is the switch itself.

Label: “Motion: full”

■ ZONE VIII · SOUND PERMISSION HUT — *Silent by default. Audio arrives only when invited.*M24 The Gatekeeper Owl *Strix silentii* [low risk · Web Audio]

Grants permission for this page to make short procedural sounds. No AudioContext exists until you press it.

Do — Create AudioContext lazily on first enable; suspend on disable; never persist “on” across visits. Enter/Space; aria-pressed.

Watch out — Audio cues must duplicate, never replace, visual and text feedback.

Reduced motion: Unchanged; the owl blinks without movement.

Label: “Sound: off”

M25 The Step-Volume Cricket *Gryllus gradatus* [watch closely · Web Audio]

Cycles the zoo’s sound level through 0 → 1 → 2 → 3. Plays a short tick at the new level if sound is on.

Do — Level 0 = muted even if Sound is on. Enter/Space; Arrow keys are not used, because this is a button, not a slider.

Watch out — If you need fine control, use input type=“range”. A cycling button is for ≤4 discrete levels.

Reduced motion: Bars change without bouncing.

Label: “Volume: 2 of 3”

M26 The Mockingbird *Mimus auditionis* [low risk · Web Audio]

Plays the zoo’s four UI sounds in sequence — success, warning, undo, navigate — with captions shown as each plays. If sound is off, it tells you how to turn it on.

Do — Respect the Sound gate and volume; captions always visible. Enter/Space.

Watch out — Each sound has a visible caption, so the information never depends on hearing.

Reduced motion: Captions appear without pulsing.

Label: “Audition the sound palette”

■ ZONE IX · MISCHIEF ENCLOSURE — Playful surprises with ethical fences: brief, optional, never trapping.

№27 The Shy Lemur <i>Lemur verecundus</i> [needs care · Pointer events]		
Opens a random specimen’s field-guide entry. With a mouse, it shyly dodges the pointer — at most twice — then stays put for good.		
Do — Dodge only for mouse; max 2 dodges per page; never move while focused. Never dodges. Enter/Space open a random specimen immediately.	Watch out — Unbounded fleeing violates basic operability. The fence is the design.	
Reduced motion: Never dodges.	Label: “Random specimen”	

№28 The Unguilty Goose <i>Anser sine culpa</i> [low risk · Timers / state machine]		
Takes you to the Sahilverse home, sahiltayal.com. First press shows the guilt-trip you would see on a manipulative site — crossed out — then a plain “Go” confirmation.		
Do — The second press always navigates; nothing is collected on the way out. Enter/Space twice.	Watch out — Exit paths must stay short and clearly labelled. Never hide “No thanks” in grey text.	
Reduced motion: No strike animation.	Label: “Leave the zoo”	

№29 The Looking-Glass Gecko <i>Gekko speculi</i> [low risk · CSS transitions]		
Flips the gallery to right-to-left direction (dir="rtl") so you can check that every specimen survives a bidirectional layout.		
Do — Only the gallery mirrors; header stays readable. Enter/Space; aria-pressed.	Watch out — Use logical CSS properties (margin-inline, inset-inline) so mirrored layouts don’t break.	
Reduced motion: Flip without animation.	Label: “Mirror the gallery”	

№30 The Tidy Gremlin <i>Gremlinus ordinans</i> [watch closely · DOM reorder]		
Shuffles the order of all specimens (useful for unbiased browsing). The label then becomes “Restore order”, which puts everything back.		
Do — Shuffle DOM order (Fisher-Yates), keep focus on the button; restore by number. Enter/Space. Reading order and visual order stay identical (DOM is reordered, not CSS order).	Watch out — Never reorder with CSS order/grid placement alone; screen-reader order would diverge from what is seen.	
Reduced motion: Rearranges instantly.	Label: “Scramble the zoo”	

PART FOUR

Reduced-motion translation table

prefers-reduced-motion: reduce is a request to remove *non-essential* motion, especially large movement, parallax, zooming and spinning. WCAG 2.2 success criterion 2.3.3 (Animation from Interactions, level AAA) asks that motion triggered by interaction can be disabled unless essential. Use this table to translate each kind of movement into a calmer equivalent that keeps its meaning.

Motion pattern	What it communicates	Reduced-motion equivalent
Magnetic drift toward pointer	"This is the primary action here."	Stronger hover colour; no displacement.
Spring overshoot / squash	"This is physical and has settled."	Instant state change; brief colour flash.
Shape morph (pill → square, sun → moon)	"The state has changed."	Swap the shape instantly; keep the text label change.
Ripple from touch point	"Here is where you touched."	Short outline or background tint on the whole control.
Press squash / stamp	"The press registered."	Background colour change on :active and .pressed.
Shake on armed / error	"Careful" or "That didn't work."	Warning colour + icon + announced text. Never shake alone.
Sliding toast	"Something happened; you can respond."	Toast appears in place; countdown shown as text or static bar.
Smooth scroll to target	"You travelled from here to there."	Instant jump plus focus on the destination heading.
Page-turn / flip	"The orientation changed."	Instant flip; announce the new direction.
Ambient looping animation	Mood, life, brand character.	Remove entirely, or pause by default with a play control.

Implementation note

A reliable pattern is a single attribute on the root element (for example `data-motion="reduced"`) computed from the OS media query and an optional, explicit in-page override. Style both states, and test both. Button Zoo's Tortoise Twin (#23) flips the entire exhibition between them so the comparison is immediate. An in-page override should not force motion on someone whose system asks for less unless they choose it explicitly.

PART FIVE

Three worked redesigns

Each example starts from a common, flawed button and redesigns it using the zoo’s anatomy. The flawed versions are composites of widespread patterns, not quotations of any particular product.

1 · The delete button

Don’t	A red trash icon with no text. Pressing it opens a modal: “Are you sure?” with a bright “Yes” and a grey “Cancel”. After deletion, nothing is recoverable.
Do	A text button “Delete draft” that acts immediately, then shows a toast “Draft deleted · Undo” for 8 seconds with a keyboard shortcut. Reserve a confirmation for truly irreversible actions, and when you use one, make both choices equally visible.
Why	Undo protects everyone without interrupting anyone. Labels name the target. Equal visual weight avoids steering. (Zoo specimens #18, #21.)

2 · The copy button

Don’t	An icon-only button. On press, the icon briefly turns green. No text changes, nothing is announced, and if the Clipboard API is blocked it fails silently.
Do	“Copy code” → “Copied ✓” for ~1.6 s, a polite live-region announcement, and a fallback that selects the text or shows it in a field when the clipboard is unavailable.
Why	Colour-only feedback is invisible to some users and to screen readers. Silent failure breaks trust more than visible failure. (Zoo specimens #01, #08, #13.)

3 · The exit button

Don’t	A large “Stay subscribed!” button and a small grey link reading “No thanks, I don’t like saving money.” (a pattern often called confirmshaming).
Do	Two equally weighted buttons with neutral labels — “Keep subscription” and “Cancel subscription” — a plain summary of what cancelling changes, and a confirmation screen with the end date.
Why	Respectful exits are part of a product’s honesty. The zoo’s Unguilty Goose (#28) parodies the guilt-trip, crosses it out, and then lets you leave.

PART SIX

The twelve-point accessible button test

Run this on every button before release. Each point lists a practical method. The site version (Button Zoo → “The twelve-point button inspection”) saves ticks in your browser and copies the list as Markdown.

#	Check	Question	How to test
1	Name	Does it have an accessible name that describes the action (and target), not just an icon?	Inspect the accessibility tree in browser dev tools; listen with a screen reader (NVDA, VoiceOver, TalkBack).
2	Role	Is it a real <button> (or <a href> for navigation), not a clickable <div>?	Search the markup for click handlers on non-interactive elements.
3	Keyboard	Can you reach it with Tab and activate it with Enter and Space? Any gesture has a single-press alternative?	Unplug the mouse. Tab, Shift+Tab, Enter, Space. Try every gesture’s alternative.
4	Focus	Is the focus indicator clearly visible ($\geq 3:1$ against adjacent colours) and never covered by sticky UI?	Tab through on light and dark themes; check focus is never hidden under sticky headers or toasts.
5	Target	Is the target at least 24×24 CSS px (WCAG 2.2 AA), ideally 44×44, with spacing from neighbours?	Measure the rendered box (zoo specimen #17 overlays sizes); check spacing between neighbours.
6	State	Is state exposed (aria-pressed / aria-expanded / aria-disabled) and also visible without colour alone?	Toggle each state and re-read the accessible name and aria-pressed/expanded; view in greyscale.
7	Feedback	Does every activation produce a visible result within ~100ms, and is important feedback announced?	Throttle the network in dev tools; time the first visible change after a press.
8	Motion	Under prefers-reduced-motion, is non-essential motion removed while all information is kept?	Turn on the OS “reduce motion” setting; compare with the site’s override (#23).
9	Contrast	Text $\geq 4.5:1$; boundaries and icons that identify the control $\geq 3:1$ in every theme and state?	Use a contrast checker on text, icon and border colours in every state (hover, focus, pressed, disabled).
10	Recovery	If the action is destructive, is there an undo or a proportionate confirmation?	Trigger the destructive action, then try to recover. Is the path obvious and keyboard-reachable?
11	Timing	Are timeouts generous, extendable or avoidable? Nothing depends on fast reactions?	Find every timeout. Can it be extended, turned off, or is it at least 20 s for essential tasks?
12	Honesty	Does the label promise exactly what happens? No hidden side effects, fake urgency or guilt?	Read the label aloud, press, and describe what happened. Did it match? Were there side effects?

A passing checklist is evidence of care, not proof of accessibility. Test with disabled people whenever you can; they find what checklists miss.

PART SEVEN

Exercises and prompts

Five exercises

- 1. The silent button.** Pick one button in a product you use daily. Write its trigger, rules, feedback and loops. Then turn on reduce-motion and a screen reader. What feedback disappeared?
- 2. One meaning, two motions.** Design a “saved” confirmation twice: once with full motion, once with none. Both must communicate the same three facts: it worked, where it went, how to undo.
- 3. The fence.** Invent a playful button behaviour. Write its fences first: maximum duration, how many times, which input types, how to stop it. Only then design the fun.
- 4. Thumb census.** Run the zoo’s target inspector (#17) on a page you built. Fix every red target; justify every amber one in writing.
- 5. The undo audit.** List every destructive action in a product. For each, decide: undo, confirmation, or neither — and write one sentence defending the choice.

Twenty-four design prompts

These are the prompts in the zoo’s shuffle deck (#11). Use one per session; sketch three answers before choosing one.

01 Design a “Save draft” button that tells the truth about whether the save reached the server.	13 Design a quantity stepper that a tired person can operate one-handed on a bus.
02 Make a delete button for a shared document that respects everyone else who uses it.	14 Create a “Generate” button for an AI tool that is honest about uncertainty and waiting time.
03 Design a copy button for a 6-digit code that works for someone reading with a screen magnifier.	15 Design a button for a kiosk in a Delhi metro station: noise, glare, hurry, many languages.
04 Create a “Play” button for a podcast that never surprises a person in a quiet library.	16 Invent a celebratory success animation that lasts under 400ms and still feels earned.
05 Invent a like button that cannot be used to measure anyone’s worth.	17 Design an emergency-stop button for a web app. What should it actually stop?
06 Design a submit button for a slow 2G connection: what happens in second 1, 5 and 20?	18 Make a “Share” button that tells you exactly what will be shared before it is shared.
07 Build a toggle that is unambiguous to someone who cannot distinguish red from green.	19 Design a toggle for dark mode whose icon still makes sense in right-to-left layouts.
08 Design a “Load more” button that lets a reader know when they have reached the end.	20 Create a “Download” button for a 300MB file that respects metered data plans.
09 Make an “Undo send” button for a message, with a delay that respects both people.	21 Design a button for a children’s learning app that never rewards rapid tapping.
10 Create a press-and-hold control, then design its no-hold alternative first.	22 Make a filter chip that announces how many results it will produce before you press it.
11 Design a “Cancel subscription” button that a regulator would admire.	23 Design a “Retry” button for a payment failure that cannot accidentally charge twice.
12 Make a button that looks pressed when it is on — and test whether anyone can tell.	24 Invent a mischievous button that becomes kinder the more someone uses it.

PART EIGHT

References, limitations, colophon

References

W3C (2023). Web Content Accessibility Guidelines (WCAG) 2.2. [w3.org/TR/WCAG22/](https://www.w3.org/TR/WCAG22/)

Relevant criteria: 1.4.11 Non-text Contrast (AA); 2.1.1 Keyboard (A); 2.2.1 Timing Adjustable (A); 2.3.3 Animation from Interactions (AAA); 2.4.7 Focus Visible (AA); 2.4.11 Focus Not Obscured (Minimum) (AA); 2.5.5 Target Size (Enhanced, 44×44, AAA); 2.5.7 Dragging Movements (AA); 2.5.8 Target Size (Minimum, 24×24, AA); 4.1.2 Name, Role, Value (A).

W3C WAI. ARIA Authoring Practices Guide — Button pattern. [w3.org/WAI/ARIA/apg/patterns/button/](https://www.w3.org/WAI/ARIA/apg/patterns/button/)

Keyboard interaction for buttons and toggle buttons; guidance on aria-pressed and stable labels.

MDN Web Docs. `prefers-reduced-motion`. developer.mozilla.org

Media query semantics and examples.

MDN Web Docs. `Navigator.vibrate()`. developer.mozilla.org

Browser support notes; not supported in Safari on iOS at the time of writing.

Saffer, D. (2013). Microinteractions: Designing with Details. O'Reilly Media.

Source of the trigger / rules / feedback / loops-and-modes anatomy.

Limitations — read before relying on this guide

- The accessibility ratings and principles are editorial judgements by the author, not results from user studies or audits.
- Snippets are teaching examples. They are tested in the Button Zoo page context; adapt and test them in yours. They are not a component library.
- The guide summarises WCAG criteria in plain words. The normative text is the standard itself; consult it for exceptions (for example, the inline-link and spacing exceptions to 2.5.8).
- Browser support for vibration, clipboard and audio APIs changes over time and differs by platform. Feature-detect; never assume.
- Specimen names, Latin binomials and all prompts are original inventions for this exhibition. Any resemblance to real products is about common patterns, not specific companies.
- No part of this guide is legal advice about accessibility or consumer-protection compliance.

Colophon

Written and designed by Sahil Tayal for Button Zoo, part of the Sahilverse (sahiltayal.com). Set in DejaVu Serif, Sans and Mono. Generated from the same data file that builds the website, so the thirty entries here match the thirty specimens online. Version 1.0 · 25 October 2026. No tracking, no sign-up — share it freely with attribution.

Return to the atlas: sahiltayal.com · **Visit the zoo:** buttonzoo.sahiltayal.online