

Elastic: Redesigning Habit Tracking

An adaptive habit tracker built for low-energy days · Anvitha Abbagani · 2026

Try the prototype: elastic.abbagani.com

Most habit trackers are built around the assumption that users show up with consistent energy. They don't. Elastic is a habit tracking system designed around that reality: capturing partial effort, reducing logging friction, and replacing punitive streaks with a forgiving momentum model. The result is a system where a low-energy day is never a reason to quit.

Full supporting detail (research, competitor analysis, system rules, the full metrics table, omitted ideas, and sources) is in the Appendix.

Table of Contents

Body

- [1. Problem](#)
- [2. Product Decisions](#)
- [3. Measuring Success](#)
- [4. Key Learnings & What's Next](#)

Appendix

- [A: Research & Validation](#)
- [B: Competitor Analysis](#)
 - [Habitica: gamified habit tracker](#)
 - [Google Tasks: simple task manager](#)
 - [Notion: highly customizable productivity workspace](#)
 - [Opportunity for Elastic](#)
- [C: Problem Definition & Prioritization](#)
- [D: System Design \(Full\)](#)
 - [Elastic Tier System \(Binary Trap\)](#)
 - [Low-friction logging \(Maintenance Burden\)](#)
 - [Fuel model \(Punishment Loop\)](#)
 - [Goals, milestones, and trajectory](#)
 - [AI Mentor and COM-B \(Insight Gap\)](#)
 - [Feedback loop](#)

[E: Omitted Ideas & What Didn't Work](#)

[Omitted ideas](#)

[What didn't work](#)

[F: Full Metrics](#)

[G: Research & Influences](#)

1. Problem

Who: People building personal habits like running, studying, coding, and creative work, who have already tried notebooks, Notion, or gamified apps like Habitica, and keep dropping off.

Where it breaks: A user sets an ambitious goal, logs consistently for one to two weeks, then hits a low-energy day. Their tracker offers two options, done or missed. A partial effort counts as a failure, one missed day resets the streak to zero, and the user stops opening the app, even though they still care about the habit.

"I had a 47-day running streak and missed one day because I was sick. Seeing it reset to zero made me not want to open the app for a week. I knew logically I hadn't failed but the app said otherwise." – 22-year-old, used three habit trackers in the past year

Research: I surveyed 16 habit tracker users and ran 5 follow-up interviews across fitness, studying, and creative work (full findings: Appendix A).

- 75% abandoned a habit after a broken streak despite still intending to continue
- 88% said partial effort counted as failure in their current tracker
- 69% had stopped tracking at least once because logging felt like too much effort
- 81% had no visibility into whether their consistency was improving or declining

Core problem: Habit systems break during low-energy periods, causing users to disengage despite continued intent. Users don't lack discipline; the systems assume consistent energy.

Four failure points, in priority order (full JTBD and rationale: Appendix C):

1. **The Binary Trap:** partial effort is invisible, so 10 minutes instead of 30 is recorded as a failure.
2. **The Maintenance Burden:** logging takes more effort than the habit, so users stop logging.
3. **The Punishment Loop:** one missed day erases visible progress.
4. **The Insight Gap:** users log effort but never learn from it. Deprioritized at first, since insights need consistent data to be useful.

Competitive gap (full analysis: Appendix B): Habitica optimizes for engagement and punishes absence. Notion optimizes for flexibility and turns logging into a chore. Google Tasks ignores habits

entirely. Strava's weekly goals are more forgiving but treat a 30-minute recovery walk and a 3-hour run as the same "active day." None of them are designed for a low-energy day.

Thesis: Habit systems should adapt to user energy rather than demand consistency from it. Instead of asking "Did you complete the task?", Elastic asks "Did you move forward today?"

2. Product Decisions

Every feature maps to one failure point; nothing is decorative (full system design: Appendix D).

Failure point	Decision	Why
Binary Trap	Elastic Tier System: every habit has Mini, Plus, and Elite	Mini is "too easy to fail," so a low-energy day still counts
Maintenance Burden	Built-in focus timer and passive browser tracking (active tabs, idle filtered out, user confirms)	Logging happens inside the work, not as an extra step or tab switch
Punishment Loop	Fuel instead of streaks, plus Hide for Today and Snooze	A missed day eases fuel down instead of resetting it; real-life interruptions aren't punished
Insight Gap	AI Mentor "Protocol Insight" with a COM-B score, plus goal-linked milestones and trajectory	Turns logs into a specific next adjustment, tied to a long-term goal

Three decisions that mattered most:

1. **Tiers, not done or missed.** Mini is the lowest viable version of a habit (one page, a 15-minute jog), Plus is a typical day, and Elite rewards high-energy days. Effort is never lost, which keeps users logging on exactly the days they used to quit.
2. **Fuel, not streaks.** Each log adds fuel by tier, and each day with no log reduces it by 10%, so one missed day takes fuel from 71% to 64% rather than to zero. Snoozed and fully hidden days don't reduce it at all. Decision: a forgiving model over a punitive one, because the broken-streak moment was where every interviewee said quitting felt justified.
3. **Browser-first, not mobile-first.** High-effort habits like coding and studying happen on desktops, and picking up a phone to log invites distraction. Keeping the timer, passive tracking, and logging in the browser removes the context switch.

What I chose not to build: social features and competition, heavy gamification layers (levels, health), and advanced customization up front. Each adds pressure or friction before the core loop is proven (details and what didn't work: Appendix E).

Prototype note: in the prototype, the Protocol Insight runs on fixed rules rather than a model. It flags over-reliance on Mini (40% or more of the last 14 days' logs) and stagnation at Plus (four or more Plus logs with no Elite), giving a baseline to compare a model against before it ships.

3. Measuring Success

North Star: Progress Velocity, the 7-day average of effort logged per day, counted in tier points (Mini 1, Plus 2, Elite 3). It measures whether users keep moving forward regardless of energy, not whether they open the app. The XP shown on each habit is per-habit feedback; tier points keep effort comparable across habits.

Secondary: Consistency Rate (days with at least a Mini), Recovery Rate (users who return after inactivity), Effort Distribution (Mini vs Plus vs Elite).

Guardrails: Time to Log (logging stays frictionless), Passive Tracking Accuracy (no inflated progress from idle time; measured by the confirm rate on detected sessions), Over-reliance on Mini (the system isn't too forgiving). Full definitions: Appendix F.

4. Key Learnings & What's Next

Key learnings:

- **Reducing friction beats increasing motivation.** The users who stayed consistent longest were the ones for whom logging took the least effort.
- **Design for real behavior, not ideal behavior.** Every interviewee described the same arc: ambitious setup, one to two good weeks, one bad day, then disengagement.
- **How a product handles failure decides retention** more than how it handles success.
- **Sequence complexity.** Fuel, tiers, and the AI Mentor only work because the core logging loop is simple enough to survive a bad day.

What's next (each phase is gated on the one before it):

- **Phase 2: Adaptive Intelligence.** The AI Mentor moves from summarizing to suggesting milestones from observed behavior. Gate: passive tracking accuracy above 80% and over-reliance on Mini below 40%.
- **Phase 3: Frictionless Input.** Voice logging and passive integrations (GitHub, productivity tools, health apps) replace manual logging as the primary input. Gate: Time to Log consistently below 10 seconds in Phase 2.
- **Phase 4: Environmental Awareness.** Wearable and health data adapt daily targets to sleep and recovery. Gate: Phase 3 passive tracking is stable.

Appendix

A: Research & Validation

Why I built this. At the beginning of 2026, I tried to become more consistent with running, coding, and reading. I tried notebooks (low friction, no insights), Notion (flexible, but repetitive manual logging turned tracking into a chore), and Habitica (engaging gamification, but heavy and unreliable). Each solved part of the problem, and none survived a low-energy day, which is the norm, not the exception.

Method. To validate whether the problem was personal or systemic, I surveyed 16 habit tracker users and conducted 5 follow-up interviews across fitness, studying, and creative work.

Finding	Share of respondents
Abandoned a habit after a broken streak despite still intending to continue	75%
Said partial effort counted as failure in their current tracker	88%
Stopped tracking at least once because logging felt like too much effort	69%
Had no visibility into whether their consistency patterns were improving or declining	81%

Two findings shifted the product direction. First, users weren't quitting because they stopped caring about the habit; they quit because the system made partial effort feel worthless. Second, the logging burden was underestimated across every tool: opening the tracker felt like a separate task, which on low-energy days was one more step to skip.

Interview insights:

"I had a 47-day running streak and missed one day because I was sick. Seeing it reset to zero made me not want to open the app for a week. I knew logically I hadn't failed but the app said otherwise." – 22-year-old, had used three different habit trackers in the past year

"I do log my habits, but only on good days. On bad days I just don't open the app because I know I won't have anything to show for it. So my data only reflects my best days, not my real patterns." – 21-year-old student, primary habits: studying and exercise

"The problem with Notion is I spent more time designing my tracker than actually using it. By the time I had it looking perfect I had no energy left to log anything." – 19-year-old, switched between Notion and paper notebooks

"I did a 20-minute walk instead of my usual 5k and Strava counted it the same. That felt fine. But my habit tracker marked it as a missed run. Those are two different products saying two different things about the same day." – 22-year-old, primary habit: running

"What I actually want is something that says 'you showed up today, even if it wasn't your best,' not something that punishes me for being human." – 19-year-old, tried Habitica, abandoned after 3 weeks

B: Competitor Analysis

Insights came from hands-on testing, user interviews, App Store reviews, and qualitative research across r/habitica, r/getdisciplined, and r/productivity. Reddit surfaced the same complaints at scale (streak resets, logging fatigue, and gamification burnout appeared across hundreds of threads unprompted), giving the analysis external validity beyond personal experience.

Habitica: gamified habit tracker

Pros	Cons
Highly engaging gamified reward system	Steep learning curve for non-gamers
Large community and social accountability features	Time spent managing the game can exceed time spent building habits
RPG-style progression makes habits feel like achievements	Overwhelming interface with too many game mechanics
Web + mobile availability	Losing health for missed habits can increase anxiety
Motivation for users who enjoy gaming	Social features can lead to negative pressure
	Requires constant internet connection
	Engagement drops once players reach higher levels
	Bugs and performance issues reported

Google Tasks: simple task manager

Pros	Cons
Easy to use and lightweight	Designed for task lists, not habits
Seamlessly integrated across the Google ecosystem	Poor support for habit repetition and streak tracking
Easy to create tasks using Google Assistant	Hard to view entire lists/habit systems at a glance
Clean UI focused on task completion	No habit analytics or progress tracking
Works well for basic reminders	Limited motivational elements
	"Feels more like a business productivity tool than personal growth"

Notion: highly customizable productivity workspace

Pros	Cons

Highly customizable habit tracking system	Steep learning curve
Can integrate with goals, projects, calendars	Requires significant manual setup
Strong visual dashboards	Easy to overbuild overly complex systems
Multiple data views (table, board, calendar)	No habit reminders
Large collection of habit tracker templates	No built-in habit analytics or reporting

Opportunity for Elastic

From Habitica	From Google Tasks	From Notion
Minimal UI	Habit streak tracking	Pre-structured habit tracking
Low cognitive load	Habit analytics	Built-in habit analytics
Habit accountability without punishment	Long-term consistency visualization	Faster daily logging
Distraction-free environment	Habit completion times	Simplified dashboards

The gap: every existing tool optimizes for either engagement (Habitica) or flexibility (Notion), but none addresses what happens when a user can't show up at full capacity. Habitica punishes absence, Notion requires manual effort that becomes a chore, and Google Tasks ignores the habit dimension entirely. Strava's weekly goals are the closest to elastic effort, but flatten effort quality. Elastic was designed for the moment all of them fail: the low-energy day.

C: Problem Definition & Prioritization

The "lack of motivation" issue was actually a mismatch between how habit systems are designed and how people behave. Users set ambitious goals assuming the same energy every day, but energy fluctuates with work, health, and external demands. When users can't meet their ideal version of a habit, binary trackers offer no flexibility, and a missed day resets progress.

Jobs To Be Done. Functionally, users want to track progress toward long-term goals without adding extra effort. Emotionally, they want to feel progress on imperfect days, have partial effort acknowledged, and keep their identity as someone who is consistent instead of being reminded of failure.

JTBD: *When my energy fluctuates, I want to continue making progress toward my goals without feeling like I've failed, so that I can stay consistent over time.*

Failure point	What breaks	Priority
---------------	-------------	----------

The Binary Trap	Partial effort is invisible. When a user runs 10 minutes instead of 30, the tracker records a failure, which is exactly what it wasn't.	1st: unrecognized effort leads to immediate disengagement
The Maintenance Burden	The act of tracking becomes the habit. When logging takes more effort than the habit, users stop logging, then stop altogether.	2nd: friction prevents sustained use
The Punishment Loop	A single missed day erases visible progress, and the system meant to build consistency becomes the reason users quit.	3rd: reduces drop-off after failure
The Insight Gap	Users log effort but never learn from it, so motivation disconnects from behavior.	Deprioritized at first: insights need consistent data

Prioritization principle: fix what makes users quit first, then optimize what helps them improve. The Binary Trap and Maintenance Burden drive churn directly; insights improve the experience but don't address the root cause.

Resulting focus: enable partial wins to sustain momentum, reduce logging friction to near zero, and replace punitive feedback loops with more resilient mechanisms. Three principles followed: **Elastic Effort** (scale based on energy), **Frictionless Tracking** (no extra work), and **Progress-Based Feedback** (not punishment).

D: System Design (Full)

Elastic Tier System (Binary Trap)

- **Mini:** the lowest viable version of a habit (e.g., reading one page or meditating for 3 minutes), designed to be "too easy to fail."
- **Plus:** the standard level of effort on a typical day.
- **Elite:** high-effort sessions that reward pushing further on high-energy days.

Low-friction logging (Maintenance Burden)

- **Passive tracking:** for knowledge-based habits like coding or studying online, a browser extension logs time on active tabs and filters out idle activity. Each detected session is shown to the user to confirm or reject before it is logged (e.g., "68 min active on github.com, 9 min idle filtered out"). The confirm rate doubles as the measure for the Passive Tracking Accuracy guardrail.
- **Built-in focus timer:** users attach a timer to a habit tier; when it ends, the tier logs itself. Every tab switch adds friction and distraction, so timing, tracking, and logging stay in one place.

Fuel model (Punishment Loop)

- Consistency is a **fuel level** that rises with effort and eases during inactivity, never resetting to zero.
- Each log adds fuel based on its tier, with smaller gains as fuel approaches 100%.
- Each day with no log reduces fuel by 10% (one missed day: 71% → 64%).

- **Hide for Today** (per habit) and **Snooze** (all goals) cover illness and outside commitments. Snoozed days, and days where every habit is hidden, don't reduce fuel.

Goals, milestones, and trajectory

- Every habit links to a long-term goal.
- **Milestones** are user-defined targets tied to a goal (e.g., three interval sessions at Elite). They move from grayscale to color and glow as they approach completion.
- Each goal shows a **trajectory**: a projected finish date based on the last 14 days of logs (frequency and tier). If it falls past the deadline, it's flagged as a signal to raise the tier or log more often.

AI Mentor and COM-B (Insight Gap)

- The AI Mentor generates a **Protocol Insight**: patterns such as over-reliance on Mini or stagnation at Plus, with a specific adjustment for each.
- In the prototype these are fixed rules: over-reliance on Mini at 40% or more of the last 14 days' logs; stagnation at Plus at four or more Plus logs with no Elite.
- A **COM-B** score, each out of five: **Capability** from the fuel gauge, **Opportunity** from how often Hide for Today is used (external constraints), **Motivation** from effort quality (Mini vs Plus/Elite).

Feedback loop

Every log, whether manual, from the timer, or passive, is categorized into a tier, converted into progress, and reflected in fuel, goal progress, and the Protocol Insight.

E: Omitted Ideas & What Didn't Work

Omitted ideas

Idea	Why it was cut
Social features (sharing, competition)	Habitica showed competition adds pressure and comparison; for many users this leads to intimidation or negative feedback from friends, increasing churn.
Heavy gamification layers	Engaging for some, but adds complexity and friction. Elastic should feel simple and intentional for a wide range of users.
Mobile-first experience	Logging on a phone invites distraction, and high-effort habits like coding and studying happen on desktops. Browser-first keeps users in their work environment.
Advanced customization early on	Too many options distract from taking action. A simple, opinionated system needs near-zero onboarding.

What didn't work

Early version	What went wrong	Change
Short-term habits only, not linked to long-term objectives	Habits felt isolated and progress lacked a clear outcome, making motivation harder to sustain	Every habit links to a goal, with milestones and trajectory
All missed logs treated the same	Users were penalized for illness or unexpected commitments, creating guilt for things outside their control	Hide for Today and Snooze
Confetti for every log, including Mini	Rewarding at first, but users stopped distinguishing minimal effort from meaningful progress. When everything is celebrated equally, nothing feels earned	Confetti for Elite only

These iterations reinforced one principle: systems should differentiate between ability, effort, and growth rather than treating all outcomes the same.

F: Full Metrics

Type	Metric	Definition	Why it matters
Key Metric	Progress Velocity	7-day average of tier points logged per day (Mini 1, Plus 2, Elite 3)	Captures whether users are consistently making long-term measured progress
Secondary	Consistency Rate	% of days with at least a Mini log	Measures if users keep showing up, even on low-energy days
Secondary	Recovery Rate	% of users who return after inactivity	Shows whether the system reduces drop-off after failure
Secondary	Effort Distribution	Ratio of Mini vs Plus vs Elite logs	Shows whether users are sustaining or burning out
Guardrail	Time to Log	Average time taken to log an action	Ensures tracking stays frictionless, not a chore
Guardrail	Passive Tracking Accuracy	% of correctly logged active sessions vs idle time	Prevents inflated or misleading progress from passive tracking
Guardrail	Over-reliance on Mini	% of users logging only Mini efforts over long periods	If users stagnate at Mini, the system may be too forgiving

G: Research & Influences

Foundational

- Atomic Habits (Clear, 2018): the concept of minimum viable habits directly informed the Mini tier design

- BJ Fogg's Tiny Habits: validated the behavioral case for scaling effort rather than demanding consistency
- Strava's weekly goal model: the closest existing approach to elastic effort, and the gap it still leaves around effort quality

Supporting sources

- Expect Failure if You Have Poor Feedback Loops: reinforced building observability into the core loop rather than treating feedback as secondary
- HelpGuide, Burnout Prevention and Recovery: informed Hide for Today and Snooze; forced rest without guilt is a retention mechanism, not a failure
- SustainabilityDirectory, The All-or-Nothing Trap: behavioral framing for why binary systems produce dropout rather than discipline
- Princeton University Press, Why Are Habits So Sticky?: grounded the fuel model in habit formation research; consistency of context matters more than consistency of effort