



Introducing context-aware monetization

The Paywall Playbook



Dieter Rappold,
Co-Founder and CEO



Cecilie Auersperg,
Marketing Lead



Introduction

The mobile subscription economy has never been more competitive. More apps, more paywalls, more teams running more experiments - and yet overall conversion benchmarks show little sign of meaningful improvement ([RevenueCat 2026](#)).

That's not because teams aren't trying. It's because they're all trying the same things. The best layouts, the most effective CTAs, the right trial lengths, the right price anchoring - all found, documented, shared and copied. The industry has gotten very good at the what. And in doing so, it has converged ([RevenueCat 2026](#)).

This playbook started with a different question: not what should the paywall show - but when should it appear? And what is actually

happening in a user's world when it does?

To answer that, ContextSDK partnered with [Vahe Baghdasaryan](#) of Tangent - one of the most experienced paywall practitioners in the industry with over 21,000 experiments behind him - combining his expertise in what converts with ContextSDK's on-device intelligence layer that detects what a user is physically doing at the moment a paywall fires.

The result is the first framework for context-aware paywall optimization. This playbook walks through how paywalls work, why they've converged, how physical context changes the equation - and how to start experimenting.

**The next unlock is timing.
It's been hiding in plain sight.**

1. Paywall 101	→ Page 5
2. The Current State - Paywalls All Look the Same	→ 11
3. The Missing Dimension - Introducing Context-Aware Paywalls	→ 15
4. The Signal Layer	→ 20
5. What Context-Aware Paywalls Could Look Like	→ 24
6. The Opportunity	→ 26
7. Conclusion	→ 28

Paywall 101

What a paywall actually is

A paywall is the moment your app asks a user to pay. It can be hard - no bypass, subscribe or leave - or soft, allowing users to dismiss it and continue with a free tier.

Hard paywalls convert significantly better short-term ([Dev Community 2026](#)), but soft paywalls can drive broader adoption over time. Neither is universally better. The right choice depends on your product.

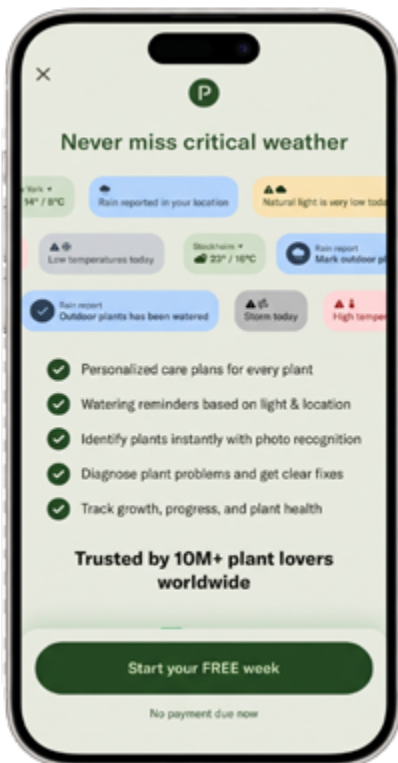
What most teams underestimate is that the paywall isn't a single screen. It's a category of touchpoints that spans the entire user life-cycle. **Vahe Baghdasaryan**, founder of [Tangent](#), has run over 21,000 paywall experiments and structures it into four distinct types.



The four types

Onboarding paywall

The onboarding paywall sits at the end of the new user flow. **Around 80 to 90% of all paid conversions happen here.** It's the highest-intent moment you'll ever have - the user just downloaded your app and wants to know if it's worth it. Everything else is downstream of getting this right.



Planta: AI Plant & Garden Care

In-app paywall

The in-app paywall appears when a user taps a locked feature or hits a plan limit. Lower conversion than onboarding, but essential for meaningful free tiers.

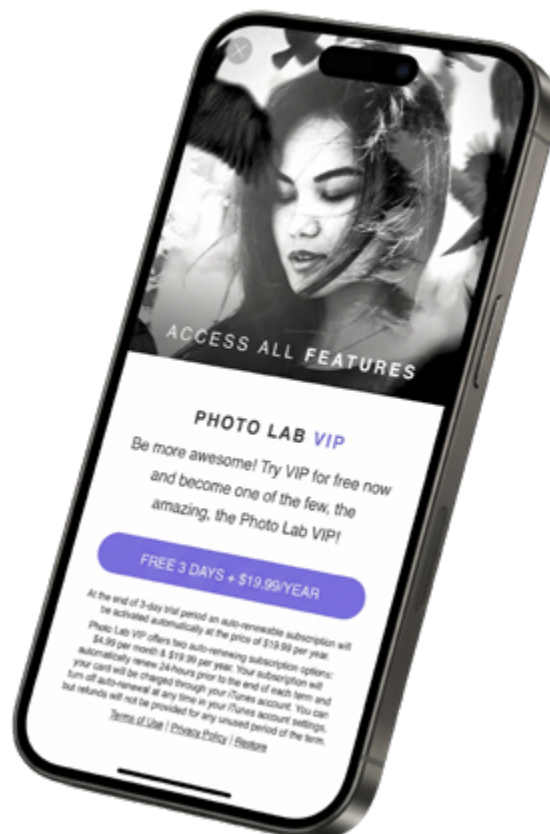


Photo Lab Editor: AI Filter

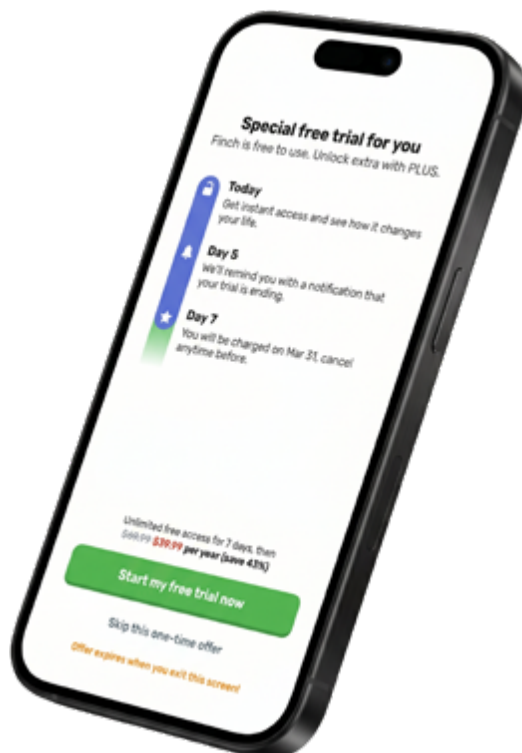


Second-time offer

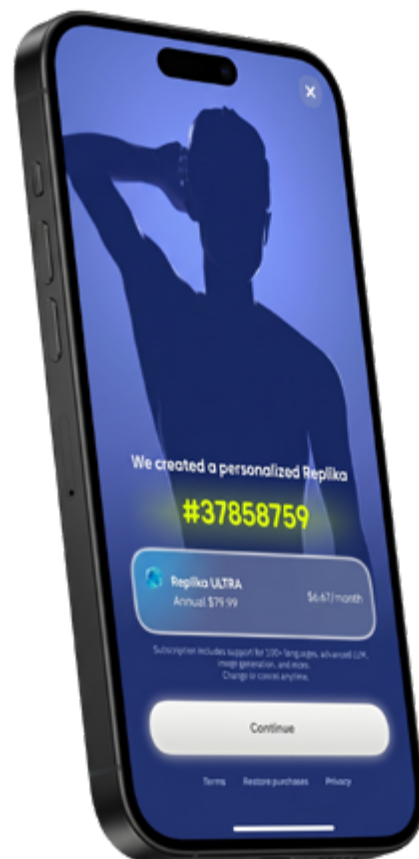
The second-time offer goes to users who dismissed onboarding and came back. A second-time user probably needs a more generous discount or a longer free trial to convert; **a transaction abandonment user already showed high intent and might only need a small nudge.**

Upsell and session paywall or win-back offer

The upsell and session paywall or win-back offer covers everything else: monthly-to-annual upgrades, win-backs, session-start placements. Win-backs work best triggered immediately after churn. **If a user cancels and you wait a week to show them an offer, it's too late. Hit them within 10 to 15 minutes, or at the next session at the latest.**



Finch: Self-Care Pet



Replika - AI Companion Chat

The four optimization levers

Value structures revenue optimization into four levers that work independently and in combination.

Paywall layout

Design, copy and conversion mechanics. **This is what most teams mean when they say “paywall optimization” - but it’s only one quarter of the picture.**

Price optimization

The right price for the right segment. **Don’t run price A/B tests in the traditional sense. Use price sensitivity surveys to generate hypotheses, test wide ranges - not \$9.99 vs \$10.99 - and evaluate on ARPU at Day 30 as a proxy for long-term LTV.**

Placement optimization

New moments in the user journey to introduce an offer. **Many teams optimize their onboarding paywall for years and never add a single additional placement.** That’s a significant missed opportunity.

Promotion optimization

The right offer for each user segment. Giving the same discount to everyone is wasteful. **Giving no discounts at all is also wasteful.** The goal is targeted conversion that maximizes blended ARPU.

What good looks like in 2026

Free trial framing is everything

Users often don’t understand a free trial is there - even after starting one. Repeat it everywhere: on the paywall, in the CTA, on the confirmation screen. “Try for \$0,” “Start my 7-day free trial” and “Claim your free trial” all outperform “Continue.” **“Continue” is now one of the least effective CTAs you can use** - despite still being the most common.

Toggle trials give users control - and it works.

Around 5 to 10% of users toggle off the trial and pay immediately - incremental revenue that wouldn’t exist otherwise.

Advanced version: price the two options differently. Trial = standard annual price. No trial = 10-25% discount. One paywall, combining layout and price optimization.

Fig. 1) Happier: Meditation

Social proof beats feature lists on iOS.

Nobody reads 15 bullet points. Three short, authentic App Store reviews with a star rating outperform any feature table. Raw and real converts better than editorial and curated. Matching review demographics to the user - same age, same goal - produces a small but consistent ARPU lift. On Android, a combination of feature list plus social proof tends to work better.

Fig. 2) YAZIO Calorie Counter & Diet



Multi-step paywalls for higher price points, not everything.

Above roughly \$80/year, users need more convincing before they see a number. A page of value, a page removing trial anxiety, then the offer. This works well for health, fitness and edtech. It doesn't work for utility apps - five screens before a paywall for a QR scanner is friction, not value.

Always break annual down to monthly.

\$59.99/year shown as “just \$4.99/month” consistently improves conversion. **Adding this anchor alone drove a 30% lift in trial start rate in parts of Latin America - with no change to the actual price.**

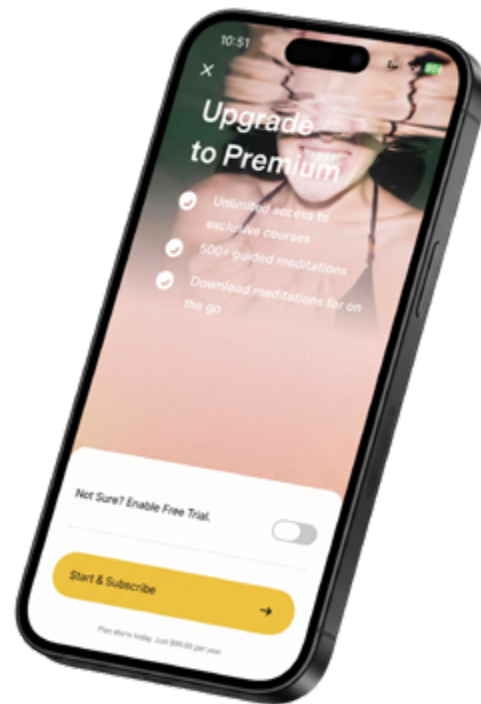


Fig. 1

Fig. 3) Calm

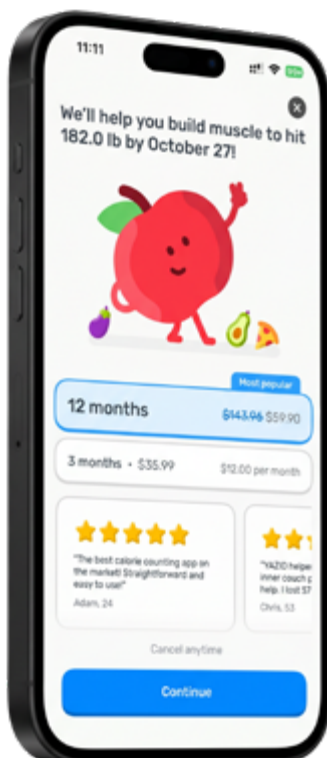


Fig. 2

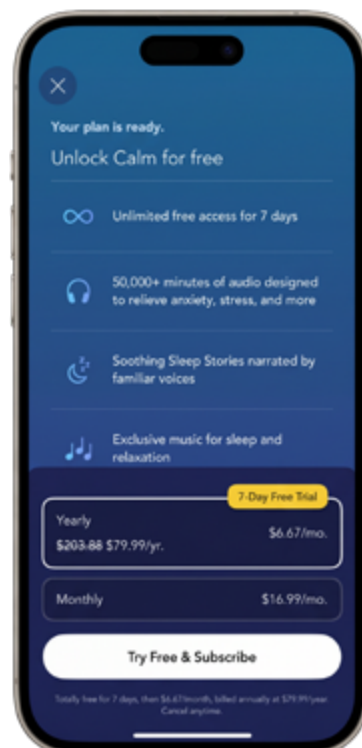


Fig. 3

Lead with the discount, not the price.

“68% off” in large text first, price second. Same offer. The first impression is “I’m getting a deal” rather than “that’s expensive.”

Fewer plan choices, higher ARPU.

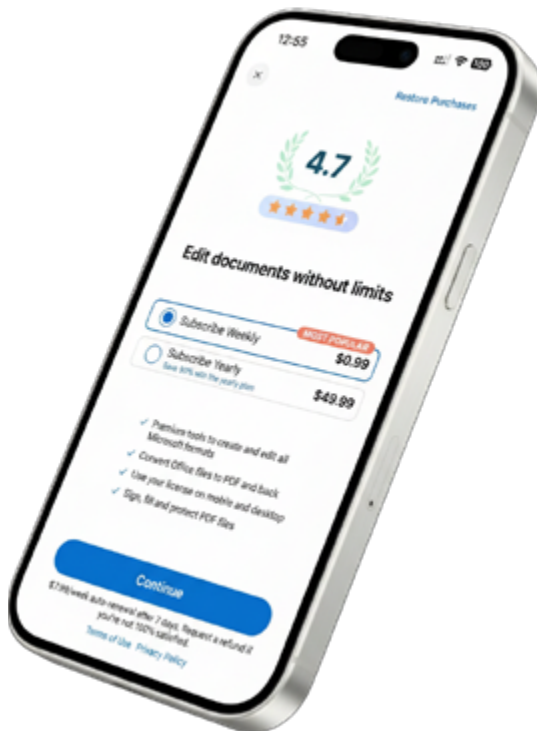
Default to annual. Put weekly behind a “view all plans” link. **Think of it like a restaurant menu. When an item is tagged “popular,” most people just order it.** Defaulting to annual can shift the annual take rate by 20% or more.

Animate discount paywalls, not necessarily standard ones.

Animated elements on promotional paywalls - a moving discount label, a spin wheel - consistently lift conversion. On standard onboarding paywalls, evidence is mixed. Test rather than assume.

At the end a paywall only needs to do two things: show the value and explain the price. FAQ sections, press logos, help buttons - these add cognitive load without adding conversion.

OfficeSuite docs & PDF editor



“You can trick the trial start. But you can’t trick the ARPU.”

Vahe Baghdasaryan

The Current State - Paywalls All Look the Same

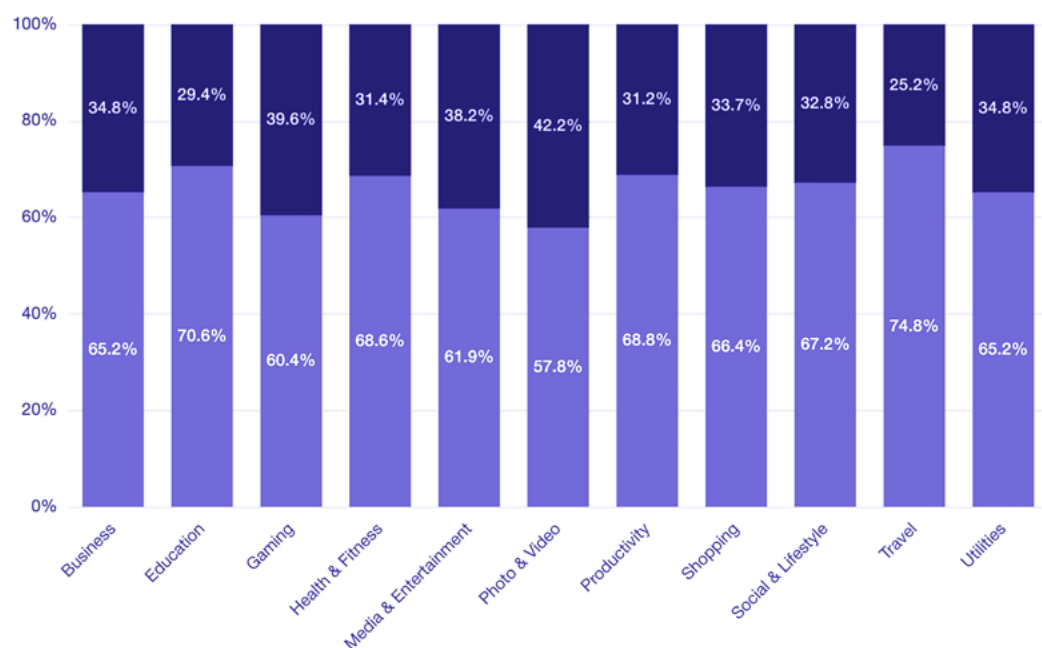
How convergence happens

The process that produces identical paywalls is completely rational. A team looks at what top-performing apps in their category are doing, implements a version of it, tests it, ships it. Every other team in the category does the same. Over a few years of iteration, the whole category converges on the same structural choices.

“Dating apps are the clearest example,” as Vahe puts it, “they’ve all A/B tested their way into the same direction - weekly plan, monthly plan, three-month plan, multiple tiers, identical layout. Not laziness. Rational imitation of what the data said worked.”

The problem is that when everyone does it, the data stops telling you anything your competitors don’t already know.

Share of scrollable paywalls by category

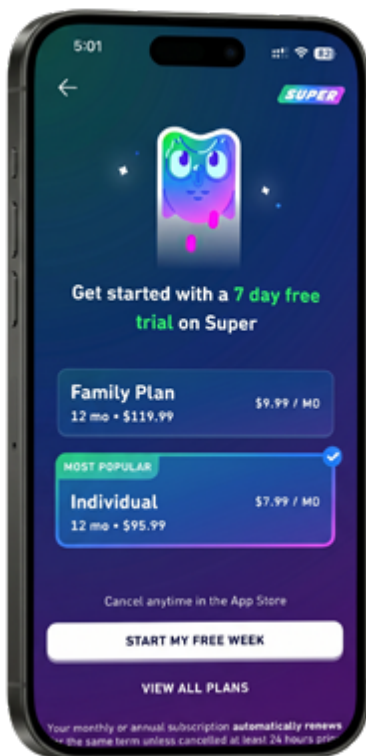


What the data shows

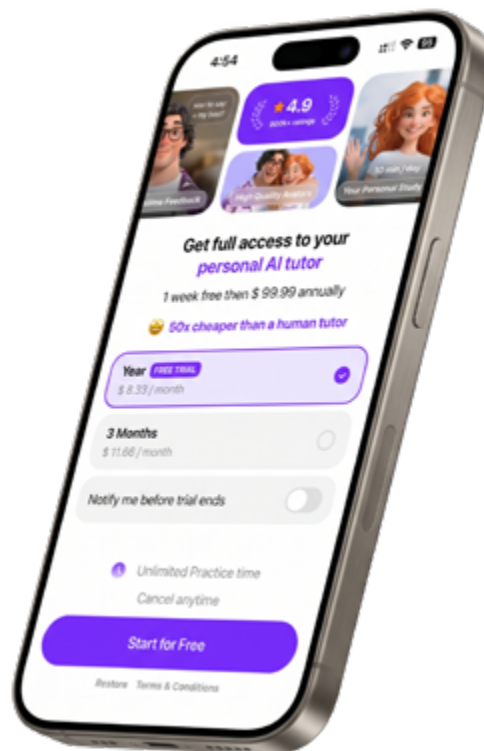
Scrollable paywalls now make up 59-76% of all paywalls across every category. High text density dominates at 45-63%. “Continue” is the most common CTA despite being one of the least effective. “Cancel anytime” is a near-universal badge ([RevenueCat 2026](#)). These patterns aren’t wrong - they reflect real learning. **But they are shared.** Showing a discount badge and “cancel anytime” has become table stakes, not a differentiator.

The measurement problem

Vahe makes the point bluntly: half the teams who think a paywall is working for them are probably measuring their A/B test wrong - looking only at trial conversion rate, not what happens to ARPU, trial cancellation rate or LTV at month six.



Duolingo:
Language & Chess



Praktika:
AI Language Tutor



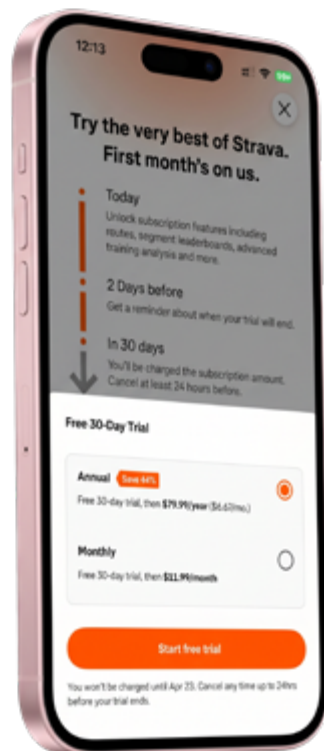
“When teams optimize for trial starts, they can win the short game and lose the long one - and the only way to know it’s happening is to measure further down the funnel than most teams do.”

- Vahe Baghdasaryan

In practice, design, copy and layout tests have a success rate of roughly 14% - compared to around 56% for pricing experiments ([Retention.Blog 2026](#)). Layout is the last place to look for leverage, yet it’s where most attention still sits.

■ The trials paradox

Trials of 17 or more days convert 70% better than short trials - 42.5% trial-to-paid versus 25.5% for trials of four days or less. And yet nearly half of all apps now use trials of four days or less ([RevenueCat 2026](#)).



Strava:
Run, Bike, Walk



Plant Parent:
Plant Care Guide

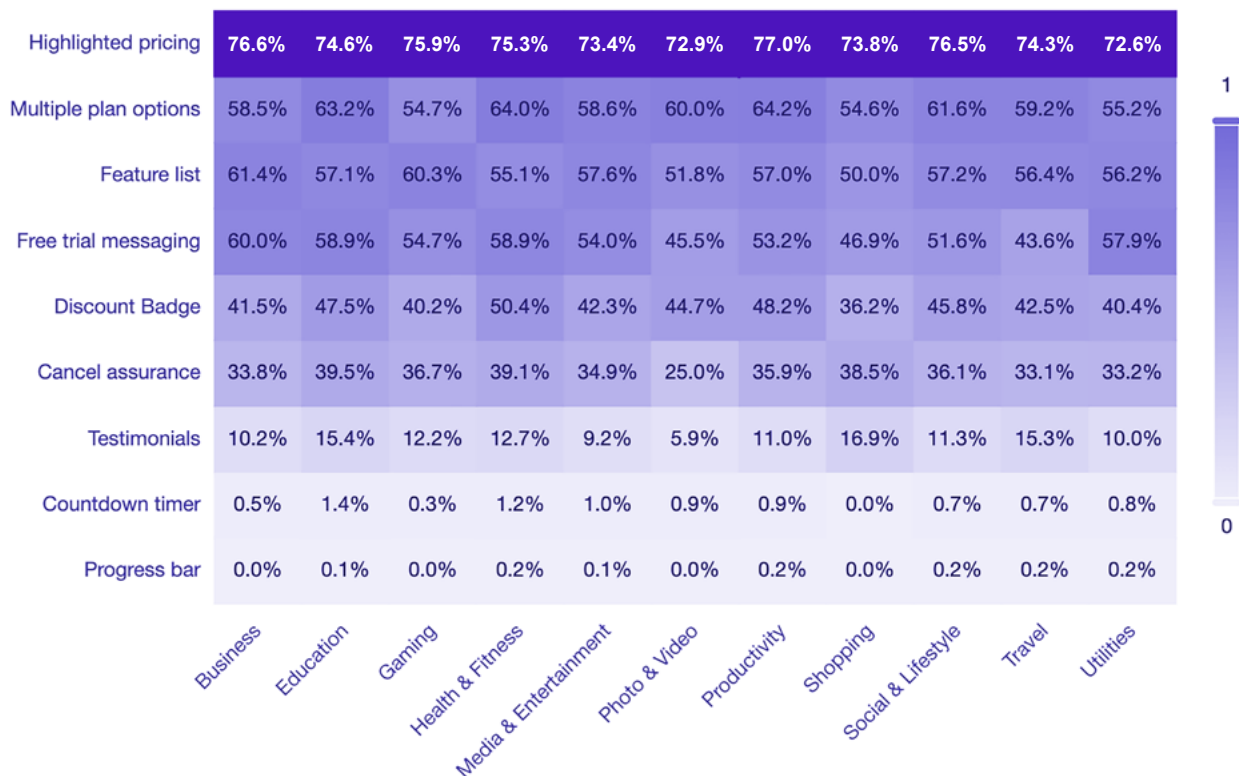
Teams are shortening trials because faster loops mean faster experiments - which is a legitimate reason. But it also means a large portion of the industry is leaving meaningful conversion on the table, knowingly, in exchange for faster iteration cycles. That's a tradeoff, not an optimization.

What hasn't been meaningfully optimized is the moment itself - not what the paywall looks like, not what price it shows, but when it appears and what is physically happening in the user's life when it does.

Where this leaves us

The paywall as a design artifact has been heavily optimized. The major structural patterns are understood, the copy frameworks are shared, the pricing tactics are documented. The design layer is no longer where the biggest gains are hiding.

Distribution of UI elements in paywalls by category



The Missing Dimension - Introducing Context-Aware Paywalls

■ The fifth dimension

Every paywall conversion is the result of five things lining up at once: the right user, seeing the right offer, at the right price, in the right placement - and at the right moment.

The industry has spent years optimizing the first four. The fifth - the actual physical moment a user encounters the paywall - has never been seriously addressed. Not because it isn't important, but because **there was no infrastructure to address it.**

Most paywalls optimize what to show. Context-aware paywalls optimize when. That's the whole thesis.

■ What the ceiling looks like

Across hundreds of optimized apps, 74.5% now use highlighted pricing, 59.2% use multiple plan options and 54% use free trial messaging ([Revenue-Cat 2026](#)). These are the structural de-

faults of a mature, converged market. The tactics aren't wrong. They're just shared. When every app is doing the same things, the question shifts: what can I test that my competitors haven't thought to try yet? Context is that thing.

Introducing ContextSDK

ContextSDK is an on-device context intelligence layer. It processes 300+ device signals through on-device Machine Learning models and outputs real-world activity states - without any new user data and without sending anything off the device.

What it detects isn't demographic data or behavioral history. It's physical reality: motion patterns, device orientation, battery level, audio environment, screen state, network type - all processed locally to answer one question: what is this person actually doing right now?

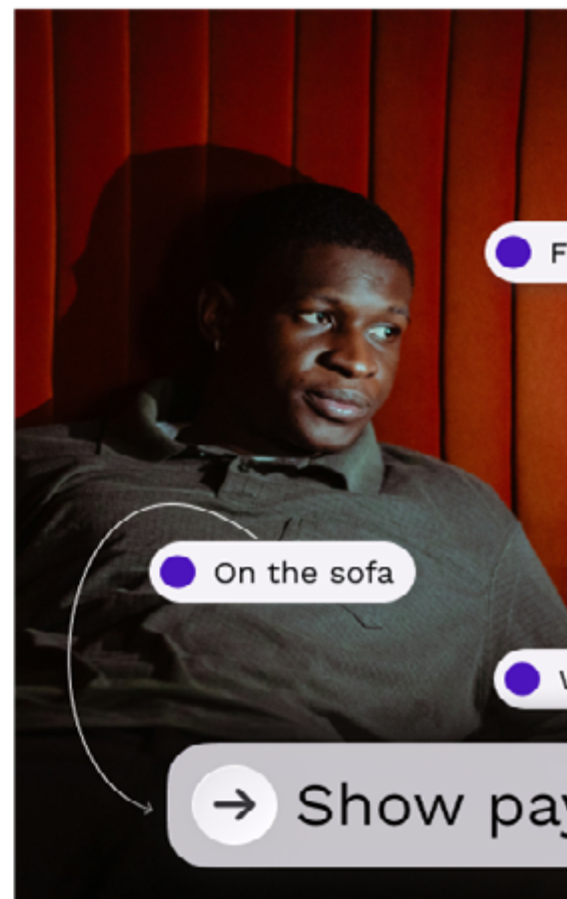
The missing dimension

The output is a context state. Stationary. Walking. Commuting. In bed. Driving. Phone on table. Phone in pocket. Each carries different implications for whether a paywall will land well or badly.

ContextSDK doesn't replace anything in your existing monetization stack. It adds a final decision layer after your segmentation rules, triggers and cooldowns have all been satisfied. Is this **actually a good moment**? If yes, show it. If no, hold it. The offer waits. The cooldown doesn't reset.

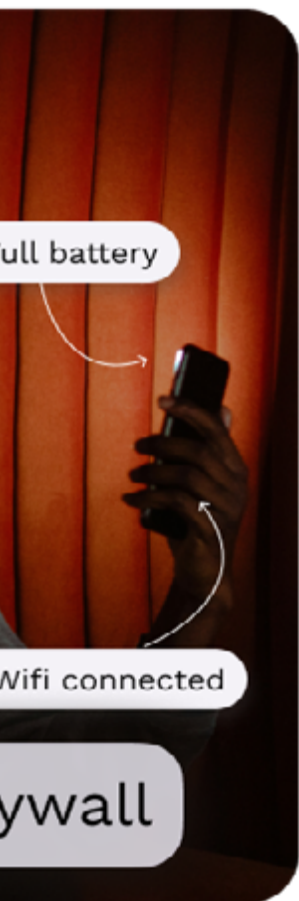
“Don't show more offers. Show them at the right moment.”

Dieter Rappold,
CEO of ContextSDK



The best paywall depends on the moment.

The collaboration: Tangent x ContextSDK



Vahe Baghdasaryan founded Tangent after years leading monetization at large consumer subscription apps. His team has built over 100 paywalls and run more than 21,000 experiments. They think in terms of user state, emotion and intent - not just layout and copy. That's exactly what's needed to make context signals actionable.

ContextSDK can tell you a user is stationary, on WiFi, at 80% battery. On its own, that's interesting. Paired with [ContextDecision's](#) understanding of what a focused, unhurried user is ready to engage with, it becomes a specific instruction: show the full multi-step paywall with an annual default and social proof. They have time. They're not going anywhere.

The driving question: "What context would make this paywall 10x more effective?" The output isn't redesigned paywalls - it's testable hypotheses.

The hypothesis format: "When user is [signal/state] → show [paywall variation] → because they're feeling [emotion] and need [trigger]."

The Signal → State → Emotion → Opportunity chain

A signal is what ContextSDK detects: stationary device, quiet surroundings, WiFi, 80% battery. A state is what those signals imply: sitting in a relaxed setting, high likelihood of intentional usage. An emotion is what the user is probably feeling: immersed, attentive, low distraction. An opportunity is the paywall action that fits: full multi-step flow, annual default, social proof. "You're in good company - join thousands already unlocking full access."

The same chain works in reverse. Phone in pocket: user not present, suppress. Phone face down: user stepped away, no offers. Battery be-

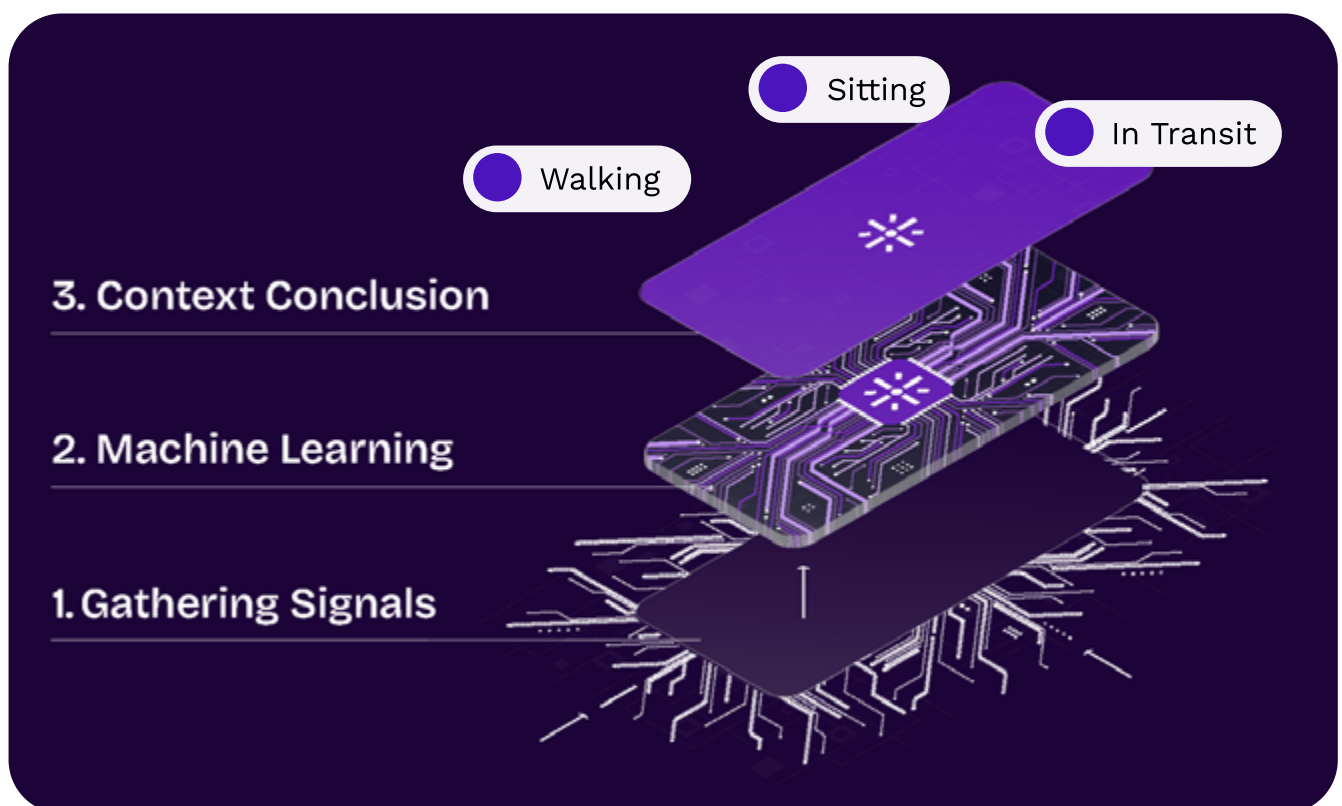
low 20%: anxious about the device, friction on friction.

The signals don't just tell you when to show a paywall. They tell you what kind of paywall the moment supports.

The downstream effects: **better moments produce better decisions**, less user fatigue and higher ARPU - because the offer that fires when someone is relaxed and focused is the annual plan, not the weekly one.

What changes - and what doesn't

Your segmentation, triggers and cooldown rules stay exactly as they are. What changes is the final gate. High-attention stationary moment + full annual paywall? Fire. Mid-commute on a shaky network + a four-screen flow? Hold.



How ContextSDK works



“The most emphasis is on the price and the duration of the subscription - when to show the weekly, when to show the annual.

If you know someone is comfortably on the sofa with a 30-minute session, you can show the yearly subscription. If they're in transit with five minutes, give them something with no commitment.”

The Signal Layer

Same app.
Same user.
Different moment.
Different ask.

■ Not features. Windows.

Context signals are windows into what a user is experiencing right now. Combined, they tell you whether this is a moment to engage or a moment to wait.

Signal reveals state.

State implies emotion.

Emotion determines opportunity.

What follows is a breakdown of every state ContextSDK detects and what each one means for how - and whether - you show a paywall.

■ The suppress logic matters as much as the show logic

Every suppressed offer in a bad moment preserves the user's patience for the next good one. The cooldown doesn't reset. The offer waits. **The user doesn't accumulate a stack of badly-timed interruptions that make them less likely to convert when the right moment finally arrives.**

The question isn't only "when would this be 10x more effective?" It's equally "when should this not be shown?" Both answers are necessary.



	Stationary	Walking	Commuting / Transit	Lying in Bed
Signal	No movement detected. Screen active. WiFi connected. Battery above 40%. Likely seated indoors.	Rhythmic step cadence detected. Accelerometer consistent with locomotion. Likely one-handed use.	Vehicle motion without driving characteristics. Movement consistent with bus, train or tram. Variable network.	Device in resting orientation. Low motion. Late timestamp. Quiet audio environment.
State	The user is settled, full attention available, in a deliberate usage session.	The user is physically moving, phone in one hand, partial attention available at best.	The user is in passive transit, phone as primary entertainment. Time is available but network may be unreliable.	The user is winding down, in a comfortable and private space. Guard is down.
Emotion	Focused. Low urgency, low distraction.	Distracted, transient. Not in a decision-making mindset.	Bored, browsing, open to discovery - but regional context shapes this significantly.	Relaxed, emotionally open. Rational decision-making is lower, emotional resonance is higher.
Paywall opportunity	Your highest-consideration moment. Show the full paywall - multi-step if warranted, annual default, social proof, toggle trial. Don't strip it down. They have time to read and decide.	Strip to a single action. No list, no scroll, no comparison view. One large tap target: "Try free."	Relaxed transit: a discovery moment. Richer layout, comparison view, social proof, limited-offer badge. High-stress commute: dial back to a single-screen prompt or suppress entirely.	Soft emotional ask. Use warm copy that connects to the user's relationship with your product. Price should be present but quiet.

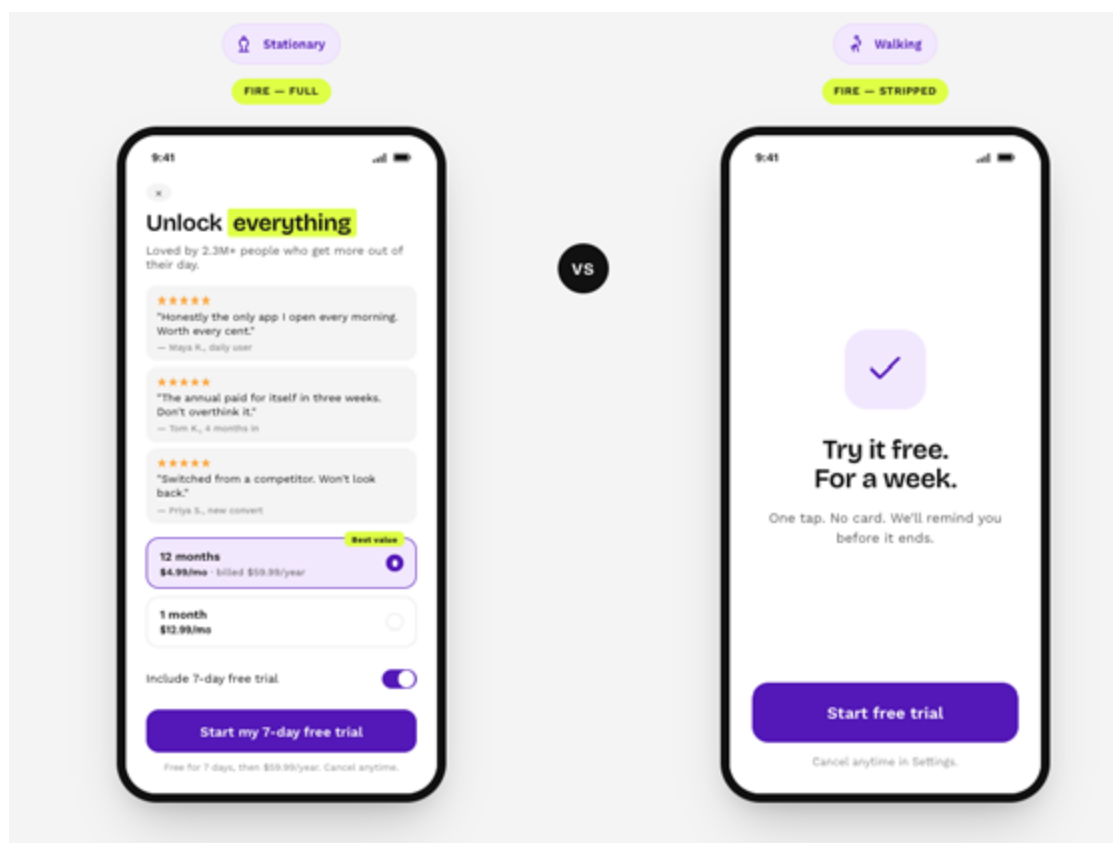
	Phone in Pocket	Battery Below 20%	Presenting to Others	Scrolling - Active vs Passive
Signal	Device fully covered. No light exposure. Movement consistent with a pocketed device.	Battery below threshold. Low power mode possibly active.	Screen mirroring active, or orientation patterns consistent with showing the device to someone else.	Scroll velocity, rhythm and interaction depth. Active: deliberate, purposeful navigation. Passive: repetitive, low-engagement swiping.
State	The user is not looking at their phone at all.	The user is mentally occupied with finding a charger and wrapping things up.	The user's screen is visible to someone else. Their only intent is to show that one thing - no interruptions welcome.	Active scroll: the user is hunting for something specific. Passive scroll: filling time with no particular intent.
Emotion	Absent - not present in the app.	Anxious, distracted.	Socially occupied, focused on the other person.	Active: curious, engaged. Passive: bored, low tolerance for interruption.
Paywall opportunity	Suppress entirely. Preserve the offer for when they're actually holding the device.	Suppress or drastically simplify. The default is to wait.	Suppress entirely.	Show during active scrolling sessions, hold during passive ones.



Airplane	Phone on Table - Face Up	Phone on Table - Face Down	Driving	Biking
No cellular signal or airplane mode active.	Device stationary. Screen facing up. No active touch.	Device stationary. Screen facing downward. No interaction.	Vehicle motion at road-appropriate speeds. Accelerometer patterns distinct from transit.	Locomotion faster and smoother than walking. Outdoor motion pattern.
Two sub-states: No network at all (cannot complete purchase) or	The user has set their phone down but is nearby and aware. Passive presence.	The user has deliberately disengaged - in conversation, in a meeting, choosing not to look.	The user is operating a vehicle. Distraction is safety-critical.	The user is cycling. Phone likely mounted or checked at a stop.
No network: disconnected, limited intent. With WiFi: relaxed, exploratory.	Passive, low-intent.	Occupied, unavailable.	Focused on the road.	Active, occupied.
No network: suppress, or surface informational premium content. With WiFi: low-pressure, single-screen offer works well.	Soft nudge only. A light in-app prompt could work. Think ambient reminder, not active ask.	Suppress entirely.	Suppress entirely, no exceptions.	No offers. Suppress.

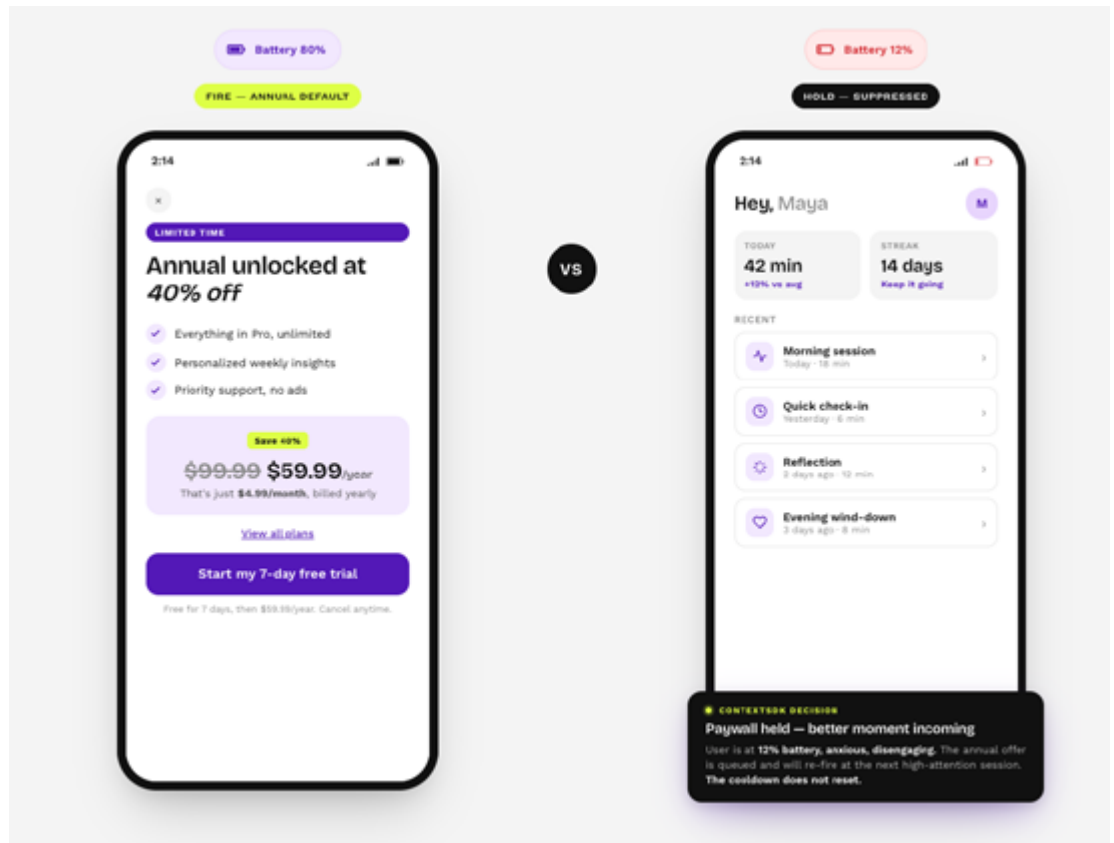
What Context-Aware Paywalls Could Look Like

01 Stationary vs Walking

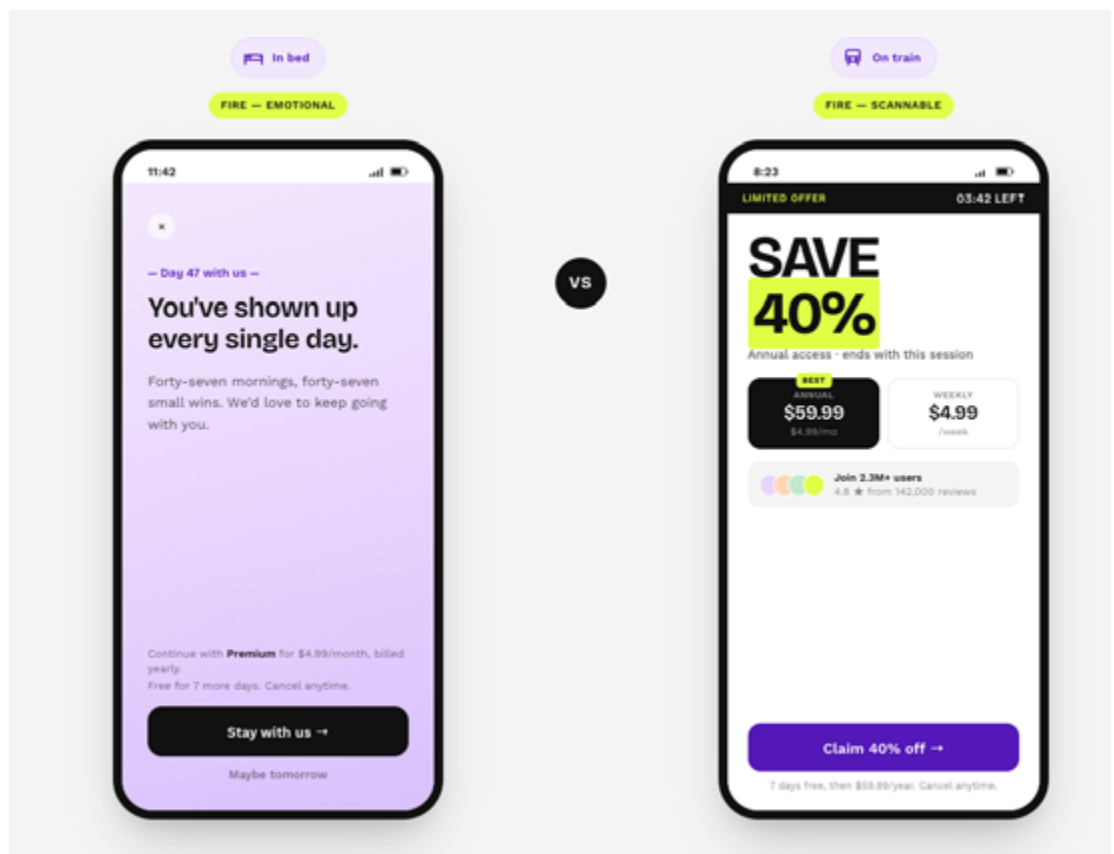




02 Battery 80% vs 12%



03 In bed vs on train



The Opportunity

■ The 95% that didn't convert

Most apps convert 4-6% of users during onboarding. That means 94-96% of users who downloaded your app, completed onboarding and saw your best paywall - didn't pay ([RevenueCat 2026](#)).

Some never would have. But a meaningful share wasn't ready at that exact moment. Walking somewhere. Battery dying. Phone face down. Half-present, in the wrong physical state to commit to something they'd just discovered.

■ They didn't say no to your product. They said no to that moment.

A significant share of total conversions happen weeks after the initial install. Those users are still there. The question is whether the paywall they encounter on Day 14 is any smarter than the one they saw on Day 0. Right now, for almost every app, the answer is no.

■ What changes when you add context

Conversion rates go up because offers appear when users are physically and emotionally positioned to engage. Same user, same app, same offer - seated at home with 40 minutes ahead of them vs standing on a platform with a train arriving. Radically different receptiveness. [Wizz by Voodoo](#) tested exactly this: by timing offers using real-world context, they captured 80% of their sales with just 20% of the prompts - a +81% conversion rate lift and +20%

LTV improvement, without touching a single line of paywall design.

User fatigue goes down because misaligned paywalls don't just fail to convert - they erode goodwill. Every bad-moment interruption is a withdrawal from the user's patience. Suppress those moments and users arrive at the good ones with more openness. The Wizz result captures this too: fewer prompts, better results. Showing less - but showing it smarter - is the whole point.

ARPU goes up because context determines which offer is shown, not just whether one is shown. Relaxed and stationary gets the annual plan. Quick-browse commute gets the weekly or a low-friction trial. The matching of offer to moment is where the revenue lift lives. [Blackbox](#) saw this directly: by triggering upsells at peak engagement moments - when players were most immersed - they achieved a +50% revenue uplift without disrupting the minimalist experience their users came for.

■ The push + paywall combination

Send a re-engagement push to a user who hasn't converted. When they tap and return, [ContextPush](#) already knows their context. Stationary at home: show the full annual paywall. On the subway: show the quick weekly offer. Driving: suppress the paywall, let them re-engage with the product.

This turns a generic re-engagement push into a moment-aware conversion sequence. The push brings the



user back. The context determines what they see when they arrive. You're not just reaching out - you're reading the room before you ask for anything.

■ Why now

The signal processing required to do this accurately, continuously, without destroying battery life or sending data off-device - that's a non-trivial engineering problem. It's now solved.

No data leaves the device. No additional permissions required beyond what the app already has. The output is a structured context state your monetization logic can act on in real time.

Because this hasn't been done before, the first movers own the learning. The hypothesis library, the design patterns, the conversion benchmarks - none of it exists yet. The teams who run these experiments first build that library for themselves.

■ Who this makes sense for

This works best for apps generating substantial monthly subscription revenue - where small percentage improvements translate into meaningful numbers. Below a certain volume, signal-to-noise is too low to draw reliable conclusions.

Beyond revenue, the best-fit apps have returning users (daily or weekly), an existing A/B testing practice and meaningful post-onboarding conversion potential. Apps where the use case spans multiple physical contexts - health, fitness, productivity, education, media - are particularly well-positioned. A fitness app is used in the gym, on the couch and on the commute. Each context implies something different about what the user

wants, and what kind of ask fits.

■ The hypothesis format

The output of this collaboration isn't redesigned paywalls. It's testable hypotheses - specific, structured predictions about what context states drive what paywall outcomes.

"When user is [signal/state] → show [paywall variation] → because they're feeling [emotion] and need [trigger]."

Four examples:

- When user is stationary with WiFi and above 20% battery → show the full multi-step annual paywall → because they have time, attention and no physical constraint on the decision.
- When user is in transit with active scrolling → show a comparison layout with a limited-offer badge → because they're in discovery mode and open to exploring something new.
- When user is in bed with a late timestamp and a long session → show a warm single-screen emotional ask with a quiet price → because rational evaluation is low and emotional connection to the product is high.
- When user is walking → suppress the paywall entirely, or reduce to a single-tap trial prompt → because they are physically unable to give the offer proper attention.

Each hypothesis is independently testable, with a clear variable, intervention and outcome to measure.

Conclusion

Design best practices spread fast. One successful experiment at a top app becomes industry standard within a year. Pricing frameworks get shared and replicated. Placement tactics diffuse across categories.

The thing that doesn't diffuse is the physical context of a user's life. That's individual, real-time and impossible to copy. No competitor can know your user is sitting quietly at home on a Sunday evening. No template accounts for the fact that the same user is on a packed commuter train on Tuesday morning with three stops to go.

Ready to test what nobody else has tested yet?

The industry has spent years getting very good at the what. The next unlock is the when. And the when is knowable, right now, for any app willing to run the experiment.

We're running our first context-aware paywall experiment this quarter - and it's free. One spot, one app, one genuine collaboration.

You bring your app and your paywall setup. We bring the context signal layer and Tangent's framework for turning those signals into testable hypotheses.

The process: a short discovery call to understand your current setup and where the gaps are. We jointly design context hypotheses specific to your app and user behavior. You implement variants using your existing paywall tooling. ContextSDK handles context detection. We measure results together - conversion rate, ARPU, trial cancel rate - properly, not just trial starts.

The goal is learning first. If the hypotheses hold, you have a new repeatable edge. If they don't, you have data nobody else in your category has, and a framework for the next test.

This is right for you if: you're generating substantial monthly subscription revenue, running active A/B tests and your users engage across multiple physical contexts throughout the day.

This is not right for you if: you're pre-PMF, conversion happens almost entirely on Day 0 or you don't have bandwidth for structured experiments alongside your roadmap.

Apply before the end of September. We kick off Q4 experiments in October!



Dieter is
here to help you
test whether
your paywall is
showing up at the
right moment.

Dieter Rappold
dieter@contextsdk.com

Text:
Cecilie Auersperg
Dieter Rappold

Art Director:
Markus Nowak

Publisher:
ContextSDK Inc.
169 Madison Avenue
STE 2895 New York
NY 10016 United States

www.contextsdk.com