Put Your Life on Autopilot

A Phone-First AI Twin for Durable Goals, Authorized Action, and Recurring Value

Neurvona | October 25, 2026

Systems design and research agenda. Not peer reviewed. No user-study or revenue results are claimed.

ABSTRACT

Many capabilities available to a technical user in an agent terminal remain inaccessible to people who primarily use a phone. We propose a personal AI twin accessed through ordinary conversation: an owner states a high-level goal, the system clarifies success, maintains a durable plan, requests bounded permissions, and reports verified progress. The unit of work is a revisioned goal contract rather than a chat turn or a single global purpose. A transport-independent execution kernel separates conversational convenience from authorization, spending, and irreversible actions. We examine jobs, customers, funding, and relationships while distinguishing actions a system can verify from outcomes requiring another person's agreement. We specify a ten-phase implementation and an evaluation program measuring reliability, owner benefit, and subscription retention. The commercial objective is long-run monthly recurring revenue subject to positive contribution margin and useful, voluntary service. Apple messaging eligibility, trust, and outcome attribution remain open dependencies, not assumed capabilities.

1. Introduction and research question

The motivating vision is simple: everyone should be able to put their life on autopilot by messaging a twin from their phone. A technically sophisticated person can already delegate work in a terminal, inspect files, approve operations, and return later. A consumer should not need to learn that interface, manage a computer at home, or translate a life objective into API calls. The research question is whether a conversational system can make such delegation accessible while preserving the control, evidence, and recoverability that consequential work requires.

An AI twin is defined operationally here as a persistent service containing owner-approved personal context, multiple explicit goals, bounded permissions, durable execution state, and receipts for actions. It is not a claim to reproduce an owner's psychology. The central interface accepts requests such as finding a job, preparing for fundraising, meeting a compatible partner, or acquiring customers. These requests are starting points for clarification. They do not independently authorize contact, payment, publication, account connection, or disclosure of personal information.

Research on reasoning with actions and on memory-based simulated agents supplies architectural precedents [1,2]. Mixed-initiative interaction and human-AI guidelines motivate interruption, correction, explanation, and termination controls [5,6]. None establishes that a personal twin can reliably improve an owner's real-world life. This paper therefore contributes a systems specification, a staged implementation, and falsifiable evaluation criteria rather than a claim of demonstrated consumer adoption.

The commercial hypothesis is that trustworthy, repeated progress across owner-chosen goals produces a durable subscription relationship. Maximizing message count or retaining people through unfinished work would optimize the wrong behavior. A successful job search may end; the product earns continued use only when the owner voluntarily delegates another useful task. The remainder of the paper connects this value hypothesis to goal contracts, execution safeguards, transport feasibility, and measurable recurring revenue.

2. Goal contracts and the personal task model

A conversation is a useful interface but an insufficient database. Important instructions can disappear in a transcript, an old approval can be mistaken for a new one, and a later message can silently change an earlier plan. The proposed durable unit is a goal contract G = (owner, objective, success criterion, scope, constraints, evidence, state, revision). The objective and success criterion remain editable owner statements. Scope records allowed tools and audiences; constraints record budget, time, privacy, and review expectations. Evidence distinguishes an assertion from an observed event.

A goal is separate from the twin's shared purpose and personal knowledge. The current Neurvona architecture has a shared mission used across channels; replacing it for each new request would interfere with unrelated work. In the proposed model, an owner can pursue a job search and prepare a funding round concurrently without rewriting either goal or changing approved campaigns. Goal-specific facts can be attached without becoming permanent global identity claims.

The initial lifecycle is deliberately small: draft, active, paused, completed, and archived. Completed or archived goals can reopen as drafts. Creating a draft records an intention. Explicitly activating a goal means tracking that intention; it is not a capability to send or spend. Later phases introduce planned steps with their own readiness, permission, approval, execution, waiting, and uncertain states. A completed step need not complete the parent goal. An owner can revise a success criterion or stop a goal while preserving a content-free event history.

Goal	System-verifiable progress	Owner or third-party outcome
Find a job	A reviewed application delivered; an interview booked	An offer made and accepted
Raise funding	A researched list reviewed; an introduction approved	An investor commits on agreed terms
Find a partner	An owner-edited profile; an agreed social plan	Mutual interest and a chosen relationship
Find customers	A relevant lead reviewed; an approved proposal delivered	A customer purchases and remains satisfied

These distinctions prevent progress reports from overstating success. Sending a hundred applications is not equivalent to finding a job, and a reply is not proof of romantic interest. Relationships remain owner-led and depend on reciprocal consent. The twin can help the owner think, prepare, and follow through; it cannot promise another person's decision or interpret a goal as permission to impersonate the owner in intimate conversations.

Phase 1 stores the owner-defined portion of the contract: objective, success criterion, next step, deadline, constraints, state, and revision. Structured permissions and execution evidence follow in later phases. It supplies a mobile goal inbox with editable fields and explicit tracking controls. Future planners can propose steps against this record without replacing the twin's shared purpose.

3. A conversation layer above a durable kernel

The proposed architecture separates five concerns: an authenticated conversation adapter, the owner-scoped goal store, a planner, an authorization service, and durable workers. The adapter accepts a message and supplies its verified owner, channel, and delivery identifier. The goal store deduplicates the event and returns a receipt. Planning may propose steps, but only an authorized executor can claim work. Progress views are reconstructed from stored state rather than from optimistic assistant prose.

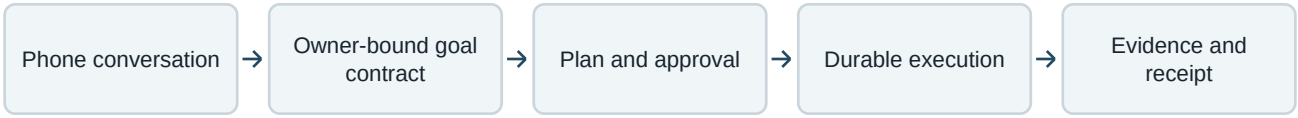


Figure 1. A channel-independent path from conversation to accountable work. Authorization remains outside generated text.

The same goal identity should survive movement between a phone web view, SMS, email, and an eligible native messaging channel. A message identifier is an idempotency key, not an authorization token. A short reply can acknowledge receipt while longer work continues asynchronously. The owner should see whether the twin is preparing, waiting for approval, waiting for resources, or awaiting reconciliation. Status and pause operations should use stored state and should not require a paid model merely to report or stop work. Existing messaging transport can still have its own charges.

Neurvona already has an authenticated web workspace, SMS and email control paths, prepaid reservations, drafts, revision-bound approval, and scheduled workers. Its owner Mac bridge currently sends approved iMessages; it does not receive owner messages or read Messages history. Consequently it cannot be represented as an existing inbound consumer twin. This audit describes source-code capabilities, not a live transport or reliability measurement.

Apple Messages for Business is a possible conversation channel through an Apple-approved Messaging Service Provider, registration, internal testing, and Experience Review [8]. It must not be described as the personal iMessage service, and its public entry point cannot be enabled before commercial approval [10]. A transport-neutral kernel allows the phone product to proceed without asserting that this dependency is already satisfied.

Apple's policies require human support during business hours and prohibit certain business models, including those allowing prepaid accounts [9]. Neurvona's credit wallet therefore raises an unresolved eligibility question. An approved provider must review the actual product and commercial flow; moving checkout elsewhere does not establish eligibility. The roadmap treats this as a go/no-go dependency and retains honestly labeled SMS and mobile web paths. No unsupported generic iMessage API or customer-owned always-on Mac is assumed.

4. Authorization, privacy, and recovery

The language model is a proposal engine, not an authority. Tool outputs, inbox messages, web pages, imported documents, and user-provided source material remain untrusted data. Prompt-injection research shows why useful task execution and resistance to hostile tool instructions must be evaluated separately [4]. The proposed authorization service evaluates structured policy outside generated text. A persuasive external document cannot grant access, increase a budget, or change an owner-approved recipient.

An approval binds an exact operation, content revision, recipient revision, account, destination, maximum charge, and expiry. A broader ongoing permission must state its channel, scope, pace, review policy, spending cap, and stop conditions. Connecting an account, subscribing, creating a goal, and approving a draft are separate actions. None should implicitly activate a different capability. A plan edit that changes recipient-visible content returns affected work to review; an irrelevant display or pacing change should not discard valid consent.

Every irreversible operation passes a single fenced claim containing the final account-standing, suppression, revision, permission, schedule, and limit checks. The durable ledger records that claim before provider dispatch. Credits are reserved before paid work and settled from known outcomes. A refusal before dispatch is free. A timeout after dispatch leaves an uncertain state and retains the reservation until provider evidence or explicit owner reconciliation resolves it. Blind retries would trade a superficially responsive chat for duplicate messages or charges.

Goal text, success criteria, and personal notes should be sealed in storage under owner- and row-specific contexts. Content-free events retain identifiers, revision numbers, action codes, and timestamps rather than duplicating intimate details into analytics. The account deletion and export map must include the goal store. A phone sender is linked through verified account pairing; possession of a phone number or a forwarded approval phrase alone is insufficient to impersonate an owner.

The goal view must show why work is blocked. In particular, waiting for research or writing credits is distinct from the owner's saved enable choice and from previously approved delivery. Adding credits may clear a resource wait on an already enabled workflow, but must never enable a paused account. Similarly, an unavailable phone transport should not erase a plan or pretend that its work has run.

Personal goals create asymmetric costs: an inappropriate romantic message, fabricated qualification, or misleading investor statement can harm someone who never used the service. Existing channel and recipient protections remain in force. The system should support preparation and owner review without claiming that a high-level objective permits unsolicited personal outreach, undisclosed commitments, or unrestricted access to other people's data.

5. Evaluation and falsifiable hypotheses

Evaluation should first establish that the system does what it claims, then test whether it helps people, and finally assess commercial value. Tool-agent benchmarks motivate checking final application state and repeating trials instead of accepting a plausible transcript [3]. Work on long software tasks further motivates measuring reliability as task duration grows, while warning that performance is domain-dependent [7]. Neither supplies a measured reliability estimate for this proposed product.

The initial offline suite should exercise owner scoping, duplicate intake, stale revision conflicts, concurrent updates, default-off flags, sealed-field handling, account export and deletion, and the absence of provider calls during goal intake. Later phases add approval expiration, changed recipients, partial provider failure, uncertain dispatch, cancelled work, resource exhaustion, and adversarial instructions embedded in retrieved content. Tests should inspect state and charges, not only the wording of a response.

Hypothesis	Measurement	Evidence that would reject it
H1: phone intake reduces setup effort	Time and steps to an owner-confirmed goal contract	No improvement against the current workspace
H2: durable goals improve continuity	Correct resumption after a day, edit, or channel switch	Lost constraints or repeated clarification
H3: bounded delegation is dependable	Repeated final-state success; unintended actions; recovery	Duplicate, unauthorized, or unreconciled work
H4: repeated benefit supports subscriptions	Goal progress, retention, net new MRR, contribution	More messages without useful progress or margin

A formative study should recruit consenting owners with concrete goals and compare the existing workflow with a phone-first contract flow. Define the task and success criterion before assignment. Instrument setup time, clarification turns, approved actions, correction effort, and owner-confirmed usefulness. Report uncertainty and attrition. Because hiring, investing, and relationships depend on outside parties, short studies should emphasize verified intermediate progress and describe final outcomes as longer-horizon observations.

A commercial pilot should use staged cohorts or randomized access when feasible. Choose a primary outcome and observation window before rollout, report enrollment and exclusions, and keep price and channel changes distinguishable. Do not infer causality from a cohort that self-selects into automation. Record activation to a useful first plan, repeat progress in later weeks, cancellation reasons, paid conversion, and contribution after model, transport, payment, and support costs.

No task-completion rate, retention uplift, willingness-to-pay estimate, or revenue gain is reported in this paper. The implementation phases below produce the instrumentation and operational prerequisites needed to test those hypotheses. Release gates establish local correctness for specified cases; they are not a substitute for a consumer study or evidence of universal real-world reliability.

6. Maximizing recurring revenue through useful progress

The business objective is to maximize expected future monthly recurring revenue, constrained by measured owner benefit, acceptable operational risk, and positive contribution margin. This objective differs from maximizing prepaid cash collected or the number of model calls. A twin that generates activity without progress can increase near-term consumption while undermining trust and renewal. Conversely, helping an owner finish a goal is success even when that goal's completion reduces subsequent usage.

MRR is monthly-normalized subscription value; our equation assumes one reporting currency. Annual subscriptions require normalization; one-time credit packs belong in a separate cash and usage analysis. New, expansion, and reactivation MRR add to the recurring base, while contraction and churn subtract from it [11]. The accounting policy must state its treatment of discounts, delinquency, owner accounts, refunds, taxes, and currency. An unadjusted contract-value estimate is not audited net recurring receipts.

$$MRR(t+1) = MRR(t) + New + Expansion + Reactivation - Contraction - Churn$$

The proposed value loop is goal intake, a concrete reviewed plan, a useful first result, visible evidence of progress, and a voluntarily chosen next step. Activation measures the owner's acceptance of a meaningful goal contract, not an initial chat message. A later progress loop should explain what happened, what is blocked, and what decision is needed. Recaps should be consensual, easy to stop, and useful without requiring the owner to maintain a dashboard.

Existing subscription catalogs and plan-first offers are a starting point. Prices and allowances should continue to come from those catalogs rather than be restated in a paper or hard-coded into an experiment. A plan can be explained in terms of supported pace and recurring service, while a one-off task remains understandable on its own. Current plan-gated-autopilot machinery exists but its launch flag is off; the paper does not assume that subscriptions are already mandatory for every active workflow.

Revenue lever	Proposed product mechanism	Guardrail
Conversion	Reach a useful reviewed plan before a contextual offer	No hidden fee or automatic enable
Retention	Show recurring, owner-confirmed progress	Do not retain people through unfinished tasks
Expansion	Offer more approved pace when it is useful	Stay within catalog prices and explicit budgets
Margin	Reuse context; cap costly work; reconcile failures	Do not charge undispatched or duplicated work

Unit economics should separate subscription and one-off revenue from model, research, messaging, payment, refund, and support costs. A subscriber whose service costs exceed contribution is not evidence of healthy growth. Weekly review should pair MRR with cohort retention, verified progress, cancellation reasons, and contribution. Which goal categories sustain recurring value remains unmeasured; acquisition spending should follow that evidence.

7. A ten-phase implementation program

The phases are ordered to make useful progress possible before expanding autonomy or distribution. Each phase has an independently reviewable result and an exit gate. New capability remains behind a default-off flag until its required dependencies and pilot evidence exist. Revenue measurement begins with the goal foundation and becomes an experiment when enough owners have experienced repeat value. Later phases are a roadmap, not a claim of completed work.

Phase	Deliverable	Exit criterion
1. Durable goal inbox	Phone-friendly goal contracts, success criteria, next steps, and revisioned lifecycle	Owner-scoped, sealed, recoverable records; no new model call or goal execution
2. Verified phone access	Pair an owner to existing messaging; assess Apple channel eligibility	Verified identity, duplicate intake handling, honest channel labels
3. Mobile plan approval	Review scope, recipients, budget, schedule, and changes on a phone	Stale approval and cross-account attempts are refused
4. Useful goal playbooks	Reuse research and drafts for jobs, customers, and funding; owner-led personal preparation	A pilot produces verified progress without inventing outcomes
5. Durable orchestration	Persist dependencies, waits, retries, and provider reconciliation	Crash recovery and repeated trials preserve at-most-one winning claim
6. Correctable goal memory	Purpose-specific facts and owner corrections across multiple goals	No cross-goal leakage or silent changes to approved facts
7. Progress and follow-through	Consensual recaps, blockers, reminders, and next-goal invitations	Repeat usefulness increases without unwanted messages
8. Recurring plan fit	Contextual offers tied to approved pace and catalog economics	Measured conversion and contribution; no surprise enable
9. Renewal and retention	Recovery, cancellation feedback, cohort analysis, and goal outcomes	Retention evidence distinguishes benefit from friction
10. Profitable distribution	Referral and acquisition experiments; capacity and support scale	Incremental MRR grows with positive measured contribution

Phase 1 intentionally does not claim inbound iMessage service, a universal planner, or autonomous execution. It builds the persistent object that all three would need. Its practical acceptance case is a person opening the workspace on a phone, recording more than one life goal, defining what progress means, setting the next step, and returning later without losing context or changing the twin's shared purpose.

Transport work can then reuse the same goal and revision identifiers instead of creating a second product inside a messaging adapter. Apple eligibility review is a dependency of an Apple channel, not of the entire goal system. If that review fails, the product can still validate useful phone delegation through honestly labeled channels before reconsidering the transport or commercial model.

8. Limitations and conclusion

The proposal does not demonstrate market demand, reliable life-goal attainment, or an increase in recurring revenue. Its architecture reduces particular failure modes but cannot guarantee model correctness, available providers, an accepted application, investment, or reciprocal affection. Phone identity can be compromised; sensitive goals require restrained collection and correction. Apple transport eligibility remains unresolved. The system also needs operational support, deletion and export coverage, and an evaluation population broader than technical founders.

The core thesis is testable: a durable, owner-controlled goal contract can turn a convenient phone conversation into accountable work. The next implementation should establish that foundation before adding broader autonomy. Neurvona's promise remains 'Put your life on autopilot'; progress toward it should be judged by useful outcomes and voluntary recurring value, not by how confidently a chat announces that it is working.

References

[1] Yao, S., et al. (2023). ReAct: Synergizing Reasoning and Acting in Language Models. ICLR.
https://arxiv.org/abs/2210.03629

[2] Park, J. S., et al. (2023). Generative Agents: Interactive Simulacra of Human Behavior. UIST, Article 2.
https://doi.org/10.1145/3586183.3606763

[3] Yao, S., Shinn, N., Razavi, P., and Narasimhan, K. (2024). tau-bench: A Benchmark for Tool-Agent-User Interaction in Real-World Domains.
https://arxiv.org/abs/2406.12045

[4] Debenedetti, E., et al. (2024). AgentDojo: A Dynamic Environment to Evaluate Prompt Injection Attacks and Defenses for LLM Agents. NeurIPS Datasets and Benchmarks.
https://arxiv.org/abs/2406.13352

[5] Horvitz, E. (1999). Principles of Mixed-Initiative User Interfaces. CHI, 159-166.
https://doi.org/10.1145/302979.303030

[6] Amershi, S., et al. (2019). Guidelines for Human-AI Interaction. CHI.
https://doi.org/10.1145/3290605.3300233

[7] Kwa, T., et al. (2025). Measuring AI Ability to Complete Long Software Tasks. NeurIPS. Current arXiv revision consulted October 25, 2026.
https://arxiv.org/abs/2503.14499

[8] Apple (2026). Apple Messages for Business. Version 3.1, July 21. Accessed October 25, 2026.
https://register.apple.com/resources/messages/messaging-documentation/

[9] Apple (2026). Apple Messages for Business: Policies. Accessed October 25, 2026.
https://register.apple.com/resources/messages/messaging-documentation/policies

[10] Apple (2026). Apple Messages for Business: Message with Your Customer. Accessed October 25, 2026.
https://register.apple.com/resources/messages/messaging-documentation/message-with-customers

[11] Stripe (2026). Billing Analytics. Metric definitions and configuration. Accessed October 25, 2026.
https://docs.stripe.com/billing/subscriptions/analytics