

Listing and lead sync into the CRM

This is a sample audit. Everything below is shown illustratively, including the data model, the screen wireframes, the integration detail, and the Phase-1 job and cost breakdown.

PREPARED FOR

[Client name], Harborstone Realty

PREPARED BY

[Business Analyst], Fastw3b

DATE

[Date]

VERSION

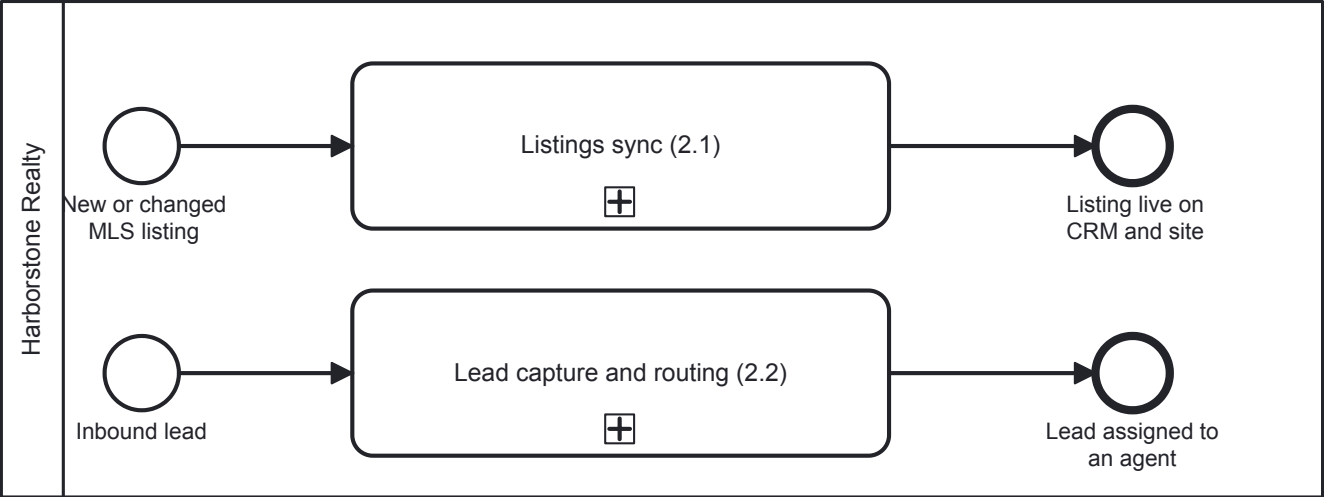
v1.0

Contents

0	Solution landscape	2
1	Executive summary	2
2	Processes	4
2.1	Listings sync	4
2.2	Lead capture and routing	5
3	Current-state diagnosis and cost of the status quo	5
	Where automation pays off most: the opportunity ranking	6
3.5	Role views	7
4	Target-state system specification	8
4.1	Information the system tracks: data model	8
4.2	Screens / views	9
4.3	Automations / functions	16
4.4	Rules & conditions	17
4.5	States / lifecycle	18
4.6	Roles & permissions: role x step x system matrix	18
4.7	Validations	20
4.8	Integrations register and data contracts	21
4.9	Edge cases and exceptions	24
4.10	Engineering coverage: the build-blocking decisions	25
5	Acceptance criteria	26
6	Phased build plan	27
7	Risks and dependencies	30
8	Non-functional expectations	31
9	What still needs confirming	32
9.1	Open questions	32
9.2	Assumptions we made (to confirm)	32
9.3	Requirements before we start	33

0. Solution landscape

The whole picture first. This automation connects two systems the brokerage already runs, the MLS and the CRM, across two everyday flows: getting listings from the MLS onto the CRM and the website, and getting inbound leads to an agent. Both land in the CRM, the single place the agents work from. The detail of each flow is in §2; what the current way costs is in §3.



Roles and dependencies.

ROLE	DOES WHAT HERE	DEPENDS ON
Listing Coordinator	Keeps listings in step between the MLS, the CRM, and the website	The MLS feed being readable, the CRM accepting listing records
Team Lead	Gets each inbound lead into the CRM and to the right agent	The website and portals delivering leads, the CRM accepting leads
Agent	Works the leads and their own listings from the CRM	Leads arriving fast and complete, listings being accurate
System (automation)	Syncs listings, captures and routes leads, notifies agents	The MLS feed, the CRM, the portals, and the notification channel

1. Executive summary

Harborstone Realty is a mid-size independent brokerage of about 20 agents. Today, a coordinator re-keys every listing from the MLS into the CRM and the website by hand, and every inbound lead is copied into the CRM and assigned from a shared inbox. Listings drift from the MLS, first contact on a lead takes hours, and some leads are never worked at all.

What this system does. It connects the two systems the brokerage already runs, the MLS and the CRM, across both flows. Listings **sync from the MLS automatically** into the CRM and the website. Inbound leads are **captured, de-duplicated, and routed to an agent by rule within seconds**, with an instant notification. The coordinator and the Team Lead stop moving data by hand and handle only the exceptions.

The value it delivers. The website stays true to the MLS with no re-keying, freeing about 8 hours a week. First contact on a lead drops from hours to minutes, which in this market is the difference between winning a client and losing them to whoever answered first, and the leads that used to leak stop leaking. Together that is worth about \$4,000 to \$7,000 a month (§3).

Recommended approach. Build both flows and their screens first (Phase 1). The §3 opportunity ranking puts the two big spots, auto-sync listings and auto-route leads, at the top, with de-duplication and the review screens riding along inside the same build. The two-way deal sync and the reporting follow once this is live.

Business logic you're approving, at a glance.

Roles (who does what):

1. The **Listing Coordinator** keeps listings in step and works the short review queue.
2. The **Team Lead** sets the routing rules and handles the leads no rule matched.
3. The **Agent** works the leads and their listings from the CRM. (The broker is a permission level of the Team Lead, with all-office reporting, not a separate workflow role. That keeps this to 3 roles.)

The flow (start to finish):

1. A new or changed MLS listing syncs into the CRM and the site automatically.
2. The coordinator reviews only the listings the feed could not place cleanly.
3. An inbound lead is captured and de-duplicated the moment it arrives.
4. The routing engine assigns it to an agent by rule and notifies them at once.
5. The agent works the lead; the Team Lead handles the few that no rule matched.

Key rules (the guardrails):

1. A listing is matched by its MLS number, then created or updated (BR-2).
2. A listing with a mapping or media problem is held for review, not published (BR-4).
3. Leads route by priority rule and round-robin, only to agents who are accepting leads (BR-8, BR-10).
4. A lead no rule matches goes to Unrouted for the Team Lead (BR-9).

5. First contact is timestamped, and an overdue lead escalates to the Team Lead (BR-12).

States (how records move):

1. A **Listing** runs *incoming to active* (or *needs review*), then *under contract*, then *sold* or *off market*.
2. A **Lead** runs *new to assigned* (or *unrouted*), then *contacted*, *working*, and *converted* or *lost*.

Edge handling (when things go sideways):

1. A slow or down feed holds and retries; captured leads are never dropped.
2. A duplicate lead is merged, not entered twice.
3. A listing that leaves the feed becomes *off market*, not deleted.

Scope: Phase 1 delivers both flows and their screens. The two-way deal and transaction sync and the reporting are later phases.

2. Processes

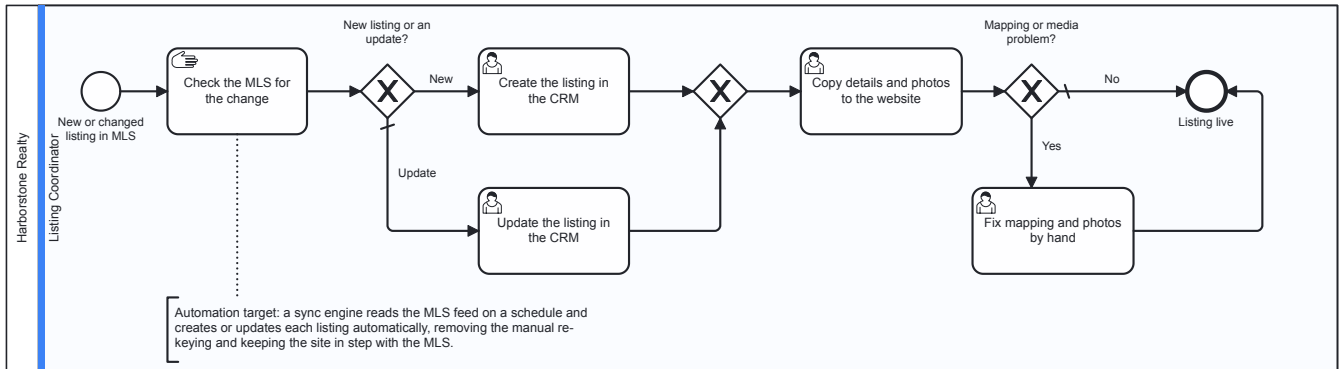
Two connected processes run the front of this brokerage: getting listings from the MLS onto the CRM and the website, and getting inbound leads to an agent. Both are done by hand today. Each current-state flow is below.

2.1 Listings sync

Roles (lanes): Listing Coordinator (watches the MLS, re-keys each listing into the CRM and the website, fixes photo and mapping problems). (*The automated target adds a System sync engine; see §4.*)

Stages (current state): (0) a listing is new or changes in the MLS, (1) the coordinator checks the MLS and re-keys it into the CRM, creating a new record or updating the existing one, (2) the coordinator copies the details and photos to the website, (3) if photos or the map are wrong, the coordinator fixes them by hand. The **automation opportunity** is annotated on the diagram.

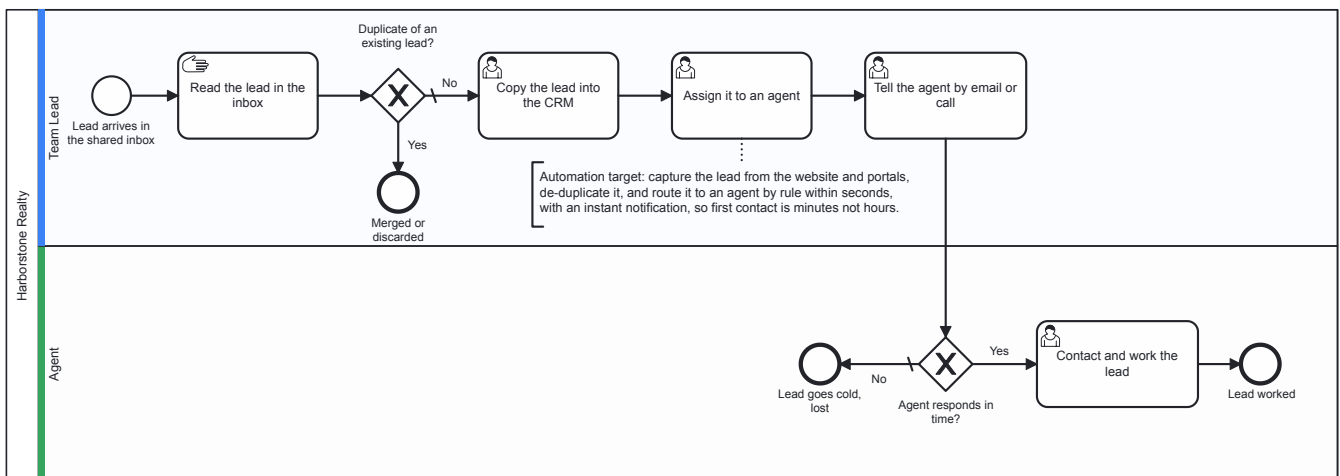
The process model below is a **real BPMN 2.0** diagram (pool with a role lane, typed start and end events, exclusive gateways with default flows, verb-noun tasks; the happy path runs straight across and the branches drop below). It renders interactively on the Process Audit page.



2.2 Lead capture and routing

Roles (lanes): **Team Lead** (reads each lead in the shared inbox, copies it into the CRM, assigns an agent, tells them) and **Agent** (contacts and works the lead). *(The automated target routes leads by rule; see §4.)*

Stages (current state): (0) a lead arrives in the shared inbox from the website or a portal, (1) the Team Lead reads it and checks whether it is a duplicate, (2) if it is new, the Team Lead copies it into the CRM, (3) assigns it to an agent and tells them, (4) the agent contacts the lead, if they see it in time. The slow, leaky first-touch is the **automation opportunity**, annotated on the diagram.



3. Current-state diagnosis and cost of the status quo

The two flows work, but both spend a person's time moving data between systems that could talk to each other, and the lead flow loses deals to whoever answers first.

- **Listings are entered twice, by hand.** The coordinator re-keys every new and changed listing from the MLS into the CRM and the website. It is about 8 hours a week of typing, and the website drifts from the MLS whenever a price or status change is missed, so buyers see stale or wrong information.

- **Leads wait in a shared inbox.** Every inbound lead is copied into the CRM and assigned by hand. First contact takes 2 to 4 hours, and overnight leads wait until morning. In this market the brokerage that contacts a lead first usually wins it, so a slow first touch is lost revenue, not just lost time.
- **Leads leak.** About 3 to 5 percent of inbound leads are never entered or worked at all, because a busy inbox loses them.
- **No single view of health.** Nobody can see at a glance which listings failed to sync, which leads are unassigned, or how fast agents respond, so problems are found late.

This is a **manual-execution** problem, not a broken process: the steps themselves are right, they are just done by a person instead of by the connection between the two systems. So the recommendation is to automate the flow as it stands, not redesign it.

Cost of the status quo. Built from the volumes in §9.2:

1. Listing re-keying: about 8 hours a week at a loaded rate of about \$35/hr is about **\$1,200/mo.**
2. Slow and leaky lead routing: on about 200 leads a month, a faster first touch and recovering the 3 to 5 percent that leak is worth conservatively one additional closed deal a quarter, about **\$3,000/mo** at an average transaction value of about \$9,000.
3. Website drift and missed updates: harder to size, but a stale site costs inquiries on the exact listings buyers are searching. Held as a **qualitative** cost, not added to the figure.

Together, about **\$4,000 to \$7,000 a month** (illustrative, built on the volume assumptions in §9.2).

Where automation pays off most: the opportunity ranking

Each spot below traces to a step in the §2 flows, ranked by effect against effort.

RANK	AUTOMATION SPOT (IN THE FLOW)	EFFECT (VALUE REMOVED)	LEVERAGE, EFFORT AND VERDICT
1	Auto-sync listings from the MLS (replaces the §2.1 re-key and publish)	Removes ~8 hrs/week of typing and keeps the site true to the MLS	Leverage: High Effort: Medium Phase 1
2	Auto-capture and route leads by rule (replaces the §2.2 copy, assign and notify)	Cuts first touch from hours to minutes and stops the 3 to 5 percent leak	Leverage: High Effort: Medium to High Phase 1
3	De-duplicate leads and listings on the way in	A clean CRM, no double records to reconcile	Leverage: Medium Effort: Low Phase 1 (folded)
4	Sync monitor and review queue for feed errors	Feed and mapping problems caught the same hour, not days later	Leverage: Medium Effort: Low Phase 1
5	Two-way deal and transaction status sync	Pipeline visibility across the CRM and a transaction tool	Leverage: Medium Effort: High Later (needs the brokerage's transaction tool, §7)
6	Reporting: lead-source return and agent response times	Spend where the good leads come from; hold agents to a response time	Leverage: Medium Effort: Medium Later

Reading. The two big wins are spots 1 and 2: they remove the two manual data-entry jobs and fix the speed-to-lead problem that costs deals. Spots 3 and 4 are low effort and ride along inside the same build, so Phase 1 is the sync engine, the lead capture and routing engine, and the screens to watch them. Spots 5 and 6 are real but depend on a tool choice or on Phase 1 data existing first, so they are later phases (§6).

3.5 Role views

A separate per-role diagram is not drawn here: the two process diagrams in §2 already place each of the three roles in its own lane across both flows, and the role-by-step matrix in §4.6 shows exactly what each role does on each screen and automation. Repeating that as three more diagrams would add nothing. What each role signs off, in one line each, is: the **Listing Coordinator** confirms the fields that must match between the MLS and the CRM and what counts as a listing that needs a human look; the **Team Lead** confirms the routing rules and the fall-back when nothing matches; the **Agent** confirms how they are notified and how fast first contact is expected.

4. Target-state system specification

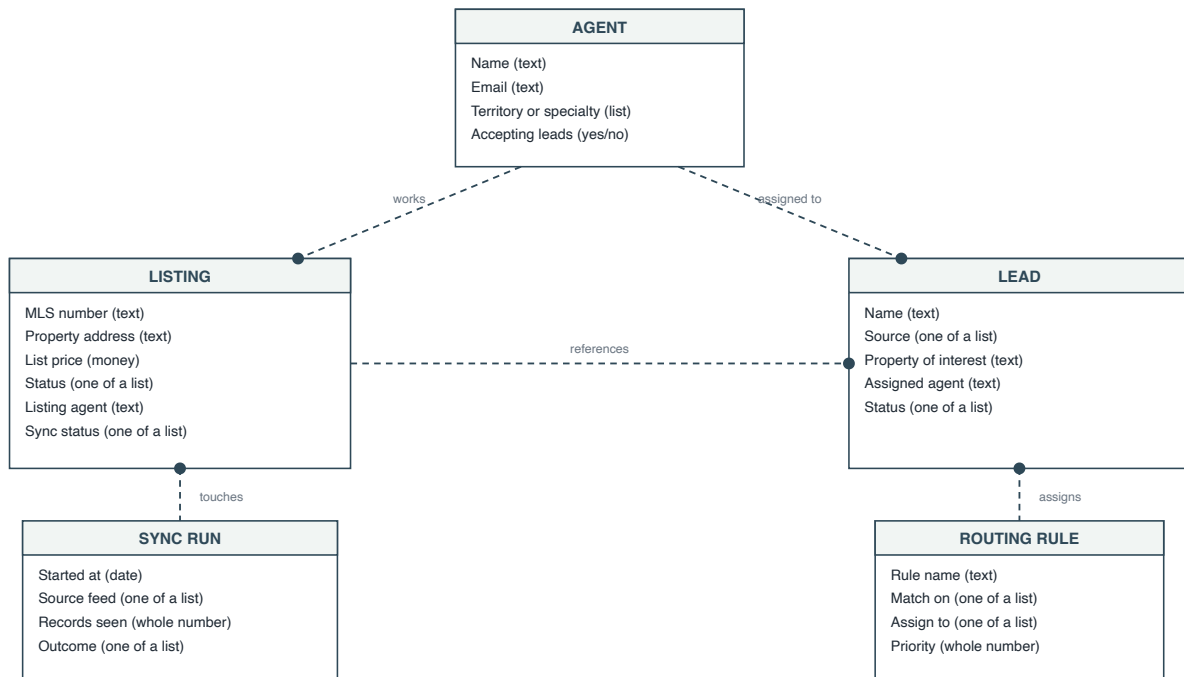
At a glance: 5 kinds of information tracked · 7 screens · 6 automations · 15 rules · 5 integrations.

4.1 Information the system tracks: data model

Each field shows the kind of information it holds, in plain terms, so you can confirm it reads right.

- **Listing.** The property record synced from the feed: MLS number (text), property address (text), list price (money), status (one of a list), bedrooms (whole number), bathrooms (number), property type (one of a list), listing agent (text), photos (files), public description (text), last synced at (date), sync status (one of a list). *(One Listing is sourced from one MLS record and is worked by one Agent.)*
- **Lead.** An inquiry from the website or a portal: name (text), email (text), phone (text), source (one of a list), property of interest (text), inquiry message (text), assigned agent (text), status (one of a list), received at (date), first contacted at (date). *(One Lead is assigned to one Agent and may reference one Listing, its property of interest.)*
- **Agent.** Name (text), email (text), phone (text), territory or specialty (one of a list), accepting leads (yes/no). *(One Agent is assigned many Leads and works many Listings.)*
- **Sync run.** A record of one pass of the feed: started at (date), source feed (one of a list), records seen (whole number), records created or updated (whole number), errors flagged (whole number), outcome (one of a list). *(One Sync run touches many Listings; a listing the run cannot place cleanly raises a review item.)*
- **Routing rule.** A rule that assigns leads: rule name (text), match on (one of a list), assign to (one of a list), priority (whole number), active (yes/no). *(Routing rules are evaluated in priority order; the first one that matches assigns the Lead to an Agent.)*

The five records and how they relate, as an **IDEF1X** data model. Every line here is a non-identifying reference (dashed, with a filled dot on the many side): a record can exist before its counterpart does, for example a Lead can sit as Unrouted with no Agent assigned yet, and a Listing can be Off market with no Sync run currently touching it.



IDEF1X data model: each entity with its key fields, and the non-identifying reference relationships between them (a filled dot marks the many side).

4.2 Screens / views

The system is seven screens across the two flows: four on the listings side for the Listing Coordinator, three on the lead side split between the Team Lead and the Agent. All seven carry the Phase-1 flow and are described and wireframed below, each with why the role needs it, what it shows, and how the person works with it. Every wireframe is drawn at a **fixed frame for its device**, the Agent's on-the-go screen on a **mobile** frame (360x780), the office screens on a **desktop** frame (1200x760), so screens of the same device read at one consistent size, and each shows the **actual fields from the data model** (not a re-imagined layout). The one item under "Also in the system" is a later-phase reporting view, not a Phase-1 screen. Wireframes show layout and fields, not the final visual design.

Listing Coordinator: Listings Sync Monitor (desktop)

Why the coordinator needs it. It is the coordinator's daily proof that the sync engine ran and what it did. Without it, the only way to know whether today's listings are current is to open the MLS and the CRM side by side and compare them by eye, which is exactly the re-keying habit this automation removes.

What they see. The **last run's** start time and source feed, its outcome, and four counts read straight off the sync run: **records seen, created, updated, and errors flagged**. A short list of recent runs sits below the counts, and a **Re-sync now** action triggers an extra run outside the schedule.

How they work with it. The coordinator glances at the outcome badge first; a clean run needs no further action. A run with errors flagged sends them straight to the Listing Review Queue to clear those items. Re-sync now is for when a feed problem was just fixed upstream and they do not want to wait for the next scheduled run.

Listings Sync MonitorListing CoordinatorDesktop

Last run today 06:00Source MLS feedCompleted with errorsRe-sync now

Records seen42

Created9

Updated31

Errors flagged2

Recent runs

WHEN	SOURCE	OUTCOME	RECORDS
Today 06:00	MLS feed	Completed with errors (2 flagged)	42
Yesterday 06:00	MLS feed	Completed cleanly	39
2 days ago 06:00	MLS feed	Completed cleanly	44

Errors flagged land in the Listing Review Queue for the coordinator to resolve.

Listings Sync Monitor (desktop, 1200x760): the coordinator's daily read on the sync engine, the last run's counts from the sync run, and a manual Re-sync now action.

Listing Coordinator: Listing Review Queue (desktop)

Why the coordinator needs it. This is where every listing the sync could not place cleanly waits for a human look. Without it, a listing with a mapping or media problem has nowhere to surface and simply never goes live, with nobody the wiser.

What they see. Every listing in Needs review, newest-first, each row showing its **MLS number**, **property address**, **list price**, and its **sync status** (the reason it was held, such as a mapping error or missing photos), with a badge for how many are waiting. A filter narrows the list by status.

How they work with it. They open each item, fix the underlying issue (correct the mapping, add or replace the photos), then resolve it, which moves the listing from Needs review to Active. The waiting count keeps an unworked backlog visible.

Listing Review Queue

Listing Coordinator

3

3 listings need a look

Filter

Needs review

MLS NUMBER	ADDRESS	PROBLEM	ACTION
MLS 4471902	128 Harbor View Dr	Photos missing	Resolve
MLS 4471880	44 Cedar Ln	Address not matched	Resolve
MLS 4471855	9 Marsh Point Rd	Price out of range	Resolve

Resolved listings go live on the CRM and the website.

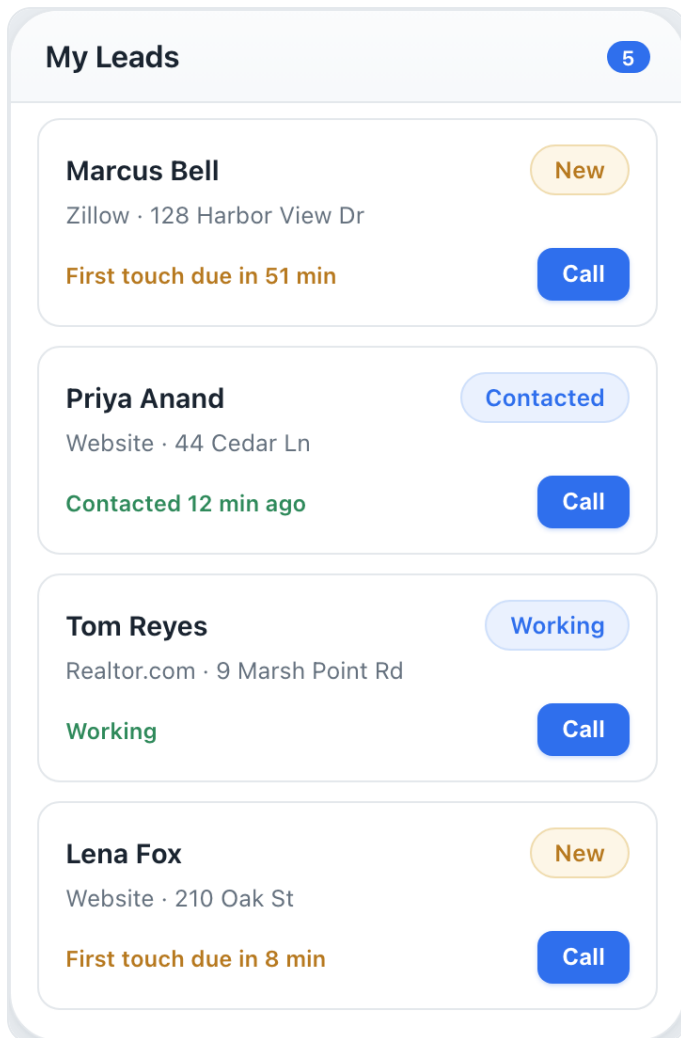
Listing Review Queue (desktop, 1200x760): every listing the sync engine could not place cleanly, with a waiting-count badge and the reason held.

Agent: Lead Inbox (mobile)

Why the agent needs it. It is the agent's way in: the leads assigned to them, in one place, already in the CRM, with no re-typing. Without it they depend on noticing a lead in a shared inbox, which is the delay this automation removes.

What they see. Their own assigned leads only, each showing the lead's **name**, **property of interest**, **source**, **received at**, and **status**, plus a first-touch timer counting down from the moment they were assigned.

How they work with it. They scan the list and tap a lead to open it, log the first contact, and move it forward (Contacted, then Working, then Converted or Lost). The timer keeps first-touch speed visible; an overdue lead escalates to the Team Lead on its own.



Lead Inbox (mobile, 360x780): the agent's own assigned leads, source, status and a first-touch timer from the Lead record.

Team Lead: Routing Rules (desktop)

Why the Team Lead needs it. This is the one place the lead-routing behaviour is controlled. Without it, every lead would need a manual look, which is the bottleneck this automation removes; with it, only the rules themselves need attention.

What they see. The list of routing rules in priority order, each showing its **rule name**, **match on** condition, **assign to** agents, **priority**, and whether it is **active**, with an editor below for the selected rule's fields.

How they work with it. They add or edit a rule, set what it matches on, which agents share it, and its priority relative to the others, then save. Only the Team Lead can make this change; a rule that is not active is skipped when leads are routed.

Routing Rules Team Lead Add rule				
Rules, in order				
RULE NAME	MATCH ON	ASSIGN TO	PRIORITY	ACTIVE
North side listings	Territory: North	Round-robin (3 agents)	1	☒
Zillow buyer leads	Source: Zillow	Priority queue (2 agents)	2	☒
Weekend overflow	Any	Team Lead	3	☐

Rules run top to bottom; the first match assigns the lead.

Routing Rules (desktop, 1200x760): the rules the routing engine evaluates by priority, and the editor the Team Lead uses to change one.

Listing Coordinator: Listing Detail (desktop, read for Agent)

Why the role needs it. It is the single accurate view of one listing, the same one the coordinator resolves from the review queue and the same one the agent reads before talking to a buyer. Without it, the coordinator checks the MLS and the CRM separately to be sure a listing is current, which is the double-checking this automation removes.

What they see. The listing's synced fields in full: **MLS number, property address, list price, status, bedrooms, bathrooms, property type, listing agent, photos, and public description**, with its **sync status** and **last synced** time shown at the top. A **Re-sync this listing** action pulls it fresh from the feed.

How they work with it. The Listing Coordinator opens a listing to confirm it read across correctly and can re-sync a single one if it looks stale. The Agent opens the same screen read-only, so they always see exactly what the coordinator sees, never a private copy that has drifted.

Listing Detail Listing Coordinator Desktop			
128 Harbor View Dr		In sync	Last synced today 06:00 Re-sync
			
MLS number	4471902	List price	\$725,000
Status	Active	Bedrooms	4
Bathrooms	2.5	Property type	Single family
Listing agent	Dana Ruiz	Sync status	In sync
Fields shown here are kept in step with the MLS.			

Listing Detail (desktop, 1200x760): one listing's full set of synced fields, its sync status and last-synced time, with a single-listing re-sync; edited by the coordinator, read-only for the agent.

Team Lead: Unrouted Leads (desktop)

Why the role needs it. This is the safety net for every lead that no routing rule could place. Without it, a lead that does not match a rule has nowhere to go and quietly falls out of the process, which is precisely the leakage this automation is meant to stop.

What they see. Every lead in Unrouted, newest-first, each row showing the lead's **name**, **source**, **property of interest**, and **received at** time, with a badge for how many are waiting and an **Assign to agent** action on each row.

How they work with it. The Team Lead picks up each unrouted lead, chooses the right agent, and assigns it, which moves the lead to Assigned and fires the same notification the routing engine would. A lead sits here only until it is assigned, so a shrinking count means the fall-back is being worked.

Unrouted Leads Team Lead				
2 leads no rule matched				
Waiting for assignment				
NAME	SOURCE	PROPERTY OF INTEREST	RECEIVED	ACTION
Jordan Pike	Realtor.com	9 Marsh Point Rd	6 min ago	<button>Assign</button>
Chris Vale	Website	(none given)	22 min ago	<button>Assign</button>
Assign a lead to route it into an agent's inbox.				

Unrouted Leads (desktop, 1200x760): the Team Lead's fall-back worklist of leads no rule matched, with the key Lead fields per row and an Assign to agent action.

Team Lead: Agent Roster and Availability (desktop)

Why the role needs it. This is where the Team Lead controls who is in the routing pool right now. Without it, a lead could route to an agent who is on vacation or otherwise unavailable, and the only fix would be to rewrite a routing rule, which is far heavier than a single toggle.

What they see. The list of agents, each showing **name**, **email**, **phone**, and **territory or specialty**, with an **Accepting-leads** toggle per agent.

How they work with it. The Team Lead flips an agent's Accepting-leads to no to take them out of the pool, and back to yes to return them, without touching any routing rule. Only an agent set to yes can be handed a routed lead, so the toggle alone reshapes who gets work.

Agent Roster and Availability Team Lead				Desktop
Agents				
AGENT	EMAIL	TERRITORY	ACCEPTING LEADS	
DR Dana Ruiz	dana@harborstone.example	North	☒	
SO Sam Okafor	sam@harborstone.example	South	☒	
PA Priya Anand	priya@harborstone.example	Waterfront	☒	
TR Tom Reyes	tom@harborstone.example	North	☐	
LF Lena Fox	lena@harborstone.example	South	☒	

Only agents accepting leads receive routed leads.

Agent Roster and Availability (desktop, 1200x760): every agent with name, email and territory, and the per-agent Accepting-leads toggle the Team Lead uses to shape the routing pool.

Also in the system

- 1. Reporting across listings and leads (Phase 2, desktop).** An all-office overview of sync health and lead outcomes over time (listings synced, leads captured, first-touch speed, conversion), open to the Team Lead at broker or owner permission. This is a later phase, listed here so the whole picture is visible, and is not specified now.

What the agent receives. The moment a lead is assigned, the agent gets an email like this:

Subject: *New lead assigned: Maria Chen (123 Birchwood Ln)*

Hi Dana,

A new lead has just been assigned to you.

Maria Chen *Property of interest: 123 Birchwood Ln Source: Zillow (portal lead) Phone: [phone on file] · Email: [email on file] Message: "Is this still available? Would love to see it this weekend."*

Please make first contact as soon as you can. This lead is due for first touch within the brokerage's agreed window; an overdue lead escalates to your Team Lead.

Open the lead in your Lead Inbox: [app link]

Harborstone Realty

4.3 Automations / functions

1. **MLS feed sync.** On a set schedule, the sync engine reads the MLS listing feed (via the RESO Web API) and creates a new listing or updates the matching existing listing by its MLS number, skipping records already caught by an earlier run so nothing is entered twice. Each run is recorded as a sync run, with its counts (records seen, created, updated, flagged) shown on the Listings Sync Monitor.
2. **Listing conflict flagging.** When the feed sync reaches a listing it cannot place cleanly (a mapping problem in the feed data, or photos that are missing, too few, or the wrong format), the system does not publish it. Instead the listing moves to Needs review and lands in the Listing Review Queue for the Listing Coordinator to resolve before it goes live.
3. **Lead capture.** The system captures inbound leads from the brokerage's IDX website and from the listing portals, Zillow and Realtor.com, as portal leads arriving by portal feed or parsed email. Each inbound lead is normalised into the standard Lead fields, then checked against existing leads and contacts by email or phone for de-duplication; a match is merged into the existing lead rather than creating a second one.
4. **Lead routing engine.** Once a lead is captured, the system evaluates the active routing rules in priority order. The first rule whose match condition fits the lead assigns it, sharing the assignment round-robin among the agents on that rule, and only to an agent whose Accepting-

leads is set to yes. A lead that no rule matches becomes Unrouted, for the Team Lead to assign by hand.

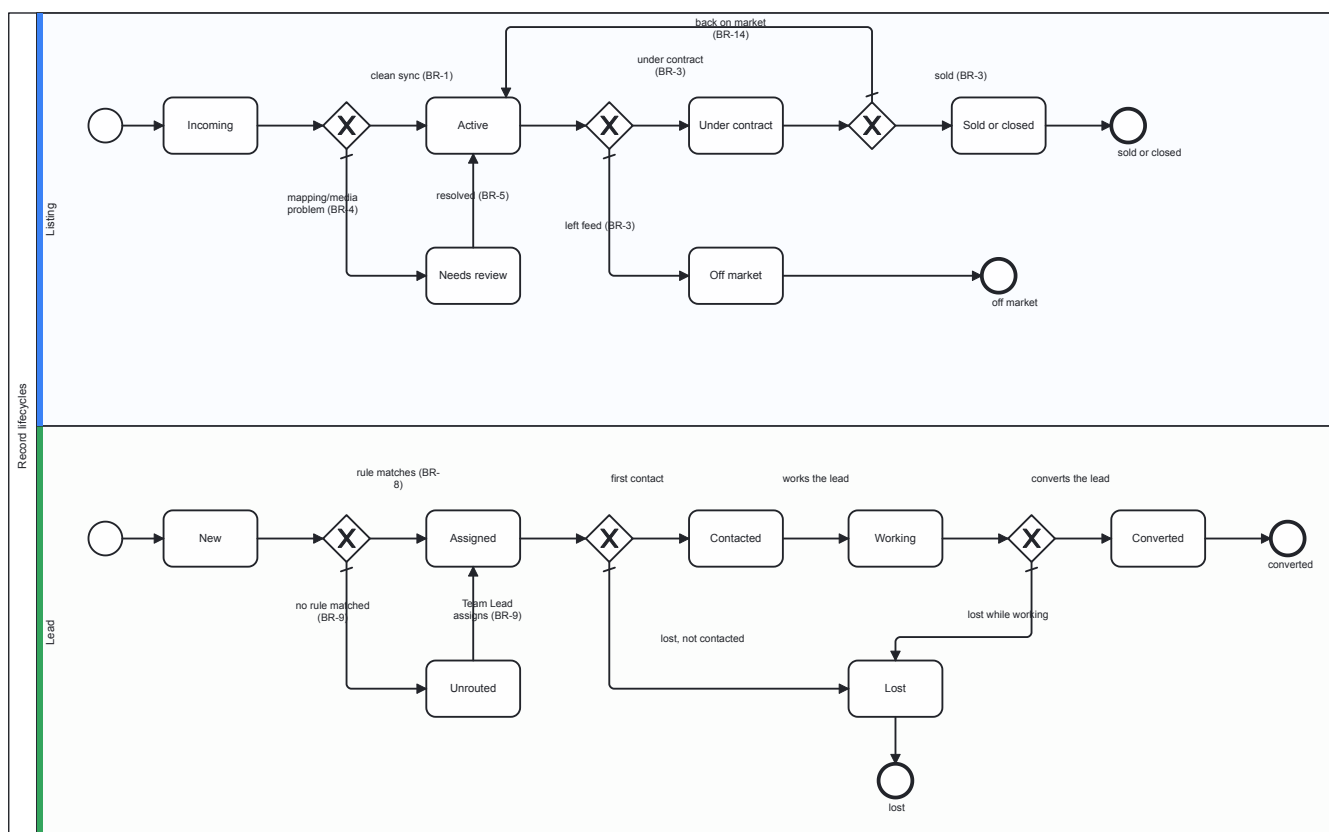
5. **Agent notification.** The moment a lead is assigned to an agent, whether by the routing engine or by the Team Lead's manual assignment, the system notifies that agent immediately by both email and text message.
6. **First-touch tracking.** The system stamps the time of the agent's first contact with an assigned lead. If that first contact has not happened within the brokerage's agreed speed-to-lead window, the system escalates the lead to the Team Lead so it does not sit unworked.

4.4 Rules & conditions

ID	RULE
BR-1	A listing that syncs cleanly from the feed becomes Active .
BR-2	A listing is matched by its MLS number: a new number creates a new listing, an existing number updates it.
BR-3	The feed's status maps directly to the listing's status: Active , Under contract , Sold or closed , or Off market .
BR-4	A listing with a mapping or media problem goes to Needs review and does not go live until the Listing Coordinator resolves it.
BR-5	Photos that are missing, too few, or the wrong format flag the listing for review.
BR-6	A lead that matches an existing lead or contact by email or phone is a duplicate: it is merged into the existing one, never entered a second time.
BR-7	A lead from a source that cannot be read automatically is still captured as it arrives and sent to Unrouted .
BR-8	Routing rules are evaluated in priority order; the first rule that matches assigns the lead, sharing the load round-robin among that rule's agents.
BR-9	When no routing rule matches, the lead becomes Unrouted and the Team Lead assigns it by hand.
BR-10	Only an agent whose Accepting-leads is set to yes can receive a routed lead.
BR-11	On assignment, the agent is notified by email and text message immediately.
BR-12	First contact with the lead is timestamped; if it is not made within the agreed time, the lead escalates to the Team Lead.
BR-13	Only the Team Lead can edit routing rules or agent availability.
BR-14	A listing that leaves the feed becomes Off market ; it is never deleted.
BR-15	A lead can reference at most one property of interest.

4.5 States / lifecycle

The two records that move through states, the Listing and the Lead, and what moves them, as one BPMN diagram (states as nodes in a lane per record). A listing enters as Incoming from the feed: a clean match becomes Active (BR-1), while a mapping or media problem sends it to Needs review instead, where the Listing Coordinator resolves it back to Active (BR-5). From Active the feed status carries the listing forward, to Under contract and on to Sold or closed, or to Off market if it leaves the feed; a listing that comes back on the market returns to Active rather than being deleted (BR-3, BR-14). A lead arrives as New: a matching routing rule assigns it straight to an agent, or, when no rule matches, it becomes Unrouted until the Team Lead assigns it by hand (BR-8, BR-9). Once Assigned, the agent's own actions carry the lead forward, to Contacted, to Working, and on to Converted, or to Lost at either the Assigned or the Working point.



4.6 Roles & permissions: role x step x system matrix

SCREEN OR AUTOMATION	LISTING COORDINATOR	TEAM LEAD	AGENT	SYSTEM
S1 Listings Sync Monitor	view + edit (trigger a re-sync)	none	none	edit (updates the run counts as it goes)
S2 Listing Review Queue	view + edit (resolve)	none	none	create (adds a listing that needs a look)

SCREEN OR AUTOMATION	LISTING COORDINATOR	TEAM LEAD	AGENT	SYSTEM
S3 Listing Detail	edit	none	view	create + edit (writes the synced fields and sync status)
S4 Lead Inbox	none	none	view + edit (work the lead, log contact)	create (populates it as leads are assigned)
S5 Unrouted Leads	none	view + edit (assign to an agent)	none	create (places a lead here when no rule matches)
S6 Routing Rules	none	view + edit	none	none (reads the active rules to route by)
S7 Agent Roster and Availability	none	view + edit	none	none (reads the Accepting-leads flag to route by)
A1 MLS feed sync	view (via S1)	none	none	create + edit (creates or updates each listing)
A2 Listing conflict flagging	view + edit (via S2)	none	none	edit (moves a listing to Needs review)
A3 Lead capture	none	view (its Unrouted output, via S5)	view (once assigned, via S4)	create (captures and de-duplicates each lead)
A4 Lead routing engine	none	edit (the rules it evaluates) + view (via S5)	none	edit (assigns the lead per the matching rule)
A5 Agent notification	none	none	view (receives the notice)	create (sends the email and text)
A6 First-touch tracking	none	view (sees an escalation)	edit (their contact stamps it)	edit (stamps first contact, escalates if overdue)

Role count. *Three process roles, Listing Coordinator, Team Lead, and Agent, plus System as the automation actor (it holds no screen of its own; it acts behind the screens above). Broker/owner is a permission level of the Team Lead role (all-office reporting), not a fourth workflow role; reporting is a later phase and is not specified here. **Step by system:** the Listing Coordinator acts at the sync side (S1 to S3, the sync engine's output), the Team Lead acts at the routing side's setup and fall-back (S5 to S7), and the Agent acts only on their own assigned leads (S4). Only the Team Lead can edit the routing rules or the agent roster's availability (BR-13); no other role has edit access to either.*

4.7 Validations

- A listing cannot move from Incoming to Active unless its MLS number, property address, list price, property type, listing agent, and at least one photo are present; missing any of these keeps it out of Active (BR-1, BR-4).
- The MLS number must be present and match the format the regional MLS issues before a listing can be matched or created (the exact format is an assumption, confirmed in [section 9](#)). A listing is matched to an existing one only by this number; no match on it means a new listing is created, never a silent duplicate (BR-2).
- Photos are checked for a minimum count and an accepted file format; a listing with photos missing, too few, or in the wrong format is held in Needs review and cannot go live until fixed (BR-5).
- The feed's status value must map to one of the four defined listing statuses (Active, Under contract, Sold or closed, Off market); a value the mapping does not recognize is not accepted silently, it raises an error on the sync run and the listing stays out of Active until reviewed (BR-3).
- A listing that drops out of the feed can only move to Off market, it is never deleted (BR-14).
- A lead needs at least one of email or phone before it can be entered; a lead with neither is captured as-is and sent to Unrouted rather than silently dropped (BR-7).
- Before a new lead is created, it is checked against existing leads and contacts by email or phone; a match merges into the existing record instead of creating a second one (BR-6, de-duplication).
- A lead can reference at most one property of interest; it is never entered against more than one listing (BR-15).
- A routing rule cannot be made active unless it has at least one agent in its assign-to list and a priority value set; a rule with no agent or no priority has nothing to round-robin among and is blocked from activation (BR-8).

- Only an agent whose Accepting-leads is set to yes can be assigned a routed lead; if every agent a matching rule points to is not accepting, the lead falls through to Unrouted rather than being assigned anyway (BR-9, BR-10).
- Routing rules and agent availability can only be changed by the Team Lead; any other role attempting the change is blocked (BR-13).
- An agent must have a valid email and phone on file before being eligible to receive routed leads, since assignment notification goes out on both channels immediately (BR-11).
- First contact cannot be timestamped before the lead's assigned timestamp; if no first contact is logged within the agreed window, the lead escalates to the Team Lead rather than staying silently overdue (BR-12).

4.8 Integrations register and data contracts

This is the heart of the automation: five outside connections, each with a plain promise on both sides of what crosses, how often, and what happens the moment it goes wrong. Nothing here is a one-off; every connection is watched, and every failure has a named landing place so nothing silently disappears.

The register at a glance

#	CONNECTS TO	DIRECTION	HOW OFTEN	ON FAILURE
I1	The MLS listing feed, read via the RESO Web API	Inbound (the listing source)	Scheduled poll, on a fixed interval	The bad record goes to Needs review (S2); the run's error count shows on the Sync Monitor (S1)
I2	The brokerage CRM	Two-way (listings and leads written in; assignment and status read back)	Out: on-event, as each listing or lead changes. In: scheduled poll, on a fixed interval	The push retries automatically; a repeated failure lands on the Sync Monitor (S1) as an error and alerts the Listing Coordinator
I3	The brokerage IDX website	Two-way (listings shown out; website inquiries captured in as leads)	Out: on-event, right after a successful sync. In: on-event, as each inquiry is submitted	Out: the site keeps showing the last good listing, nothing is pulled down on a failed push. In: an inquiry that cannot be read cleanly is still captured and sent to Unrouted
I4	Listing portals, Zillow and Realtor.com	Inbound (leads only)	By portal feed: scheduled poll. By parsed email: on-event, as each message arrives	An unreadable or unparseable lead is still captured as-is and sent to Unrouted, never dropped
I5	The agent notification channel, email and text message	Outbound	On-event, immediately on assignment	A failed send is retried once; if it still fails, the Team Lead is alerted so the agent is not left unaware of a lead they were never actually notified about

I1. The MLS listing feed, via the RESO Web API What crosses (in): the MLS number, property address, list price (money), bedrooms (whole-number), bathrooms (number), property type (one-of-a-list), listing agent, photos, the public description, and the feed's own listing status. **The data contract:** the MLS promises every active, changed, and withdrawn record is present in the feed the sync engine reads on its schedule, keyed by MLS number so the same listing is never mistaken for a new one. The brokerage's system promises to read every record on that schedule, match it by MLS number (create if new, update otherwise, per BR-2), and never delete a listing the feed drops, only mark it Off market (BR-14). A record with a mapping or media problem, for example photos that are missing, too few, or the wrong format, does not go live; it is held at Needs review until the Listing Coordinator resolves it (BR-4, BR-5).

I2. The brokerage CRM What crosses (out): the synced listing (the same fields as I1) and every captured lead (name, email, phone, source, property of interest, inquiry message, received-at, and the assigned agent once one is set), written in as each record is created or changes. What crosses (in): whichever agent the CRM shows as responsible for a lead, and any status change an agent records there (for example moving a lead to Contacted or Working), read back on the next

scheduled check. **The data contract:** the brokerage's system promises the CRM always holds the current, de-duplicated version of every listing and lead, so agents working inside the CRM are never looking at stale or doubled records (BR-6). The CRM promises to surface, in return, who currently owns each lead and what stage it is at, so the two systems stay in agreement without either side needing to poll faster than the agreed schedule.

13. The brokerage IDX website What crosses (out): the public listing fields, address, price, beds, baths, property type, photos, description, and status, refreshed right after each successful sync so the site never shows a listing the MLS has already changed. What crosses (in): a website inquiry becomes a lead: name, email, phone, source (recorded as website), property of interest, the inquiry message, and received-at. **The data contract:** the website promises to publish only what the system pushes to it, and to pass every inquiry back immediately, however the visitor typed it. The system promises the listing shown is never older than the last successful sync, and that every inquiry, even a message the site cannot format neatly, is still captured as a lead and never lost; a source that cannot be read automatically is captured as-is and sent to Unrouted rather than dropped (BR-7).

14. Listing portals: Zillow and Realtor.com What crosses (in): the same lead fields as I3, arriving either as a structured feed entry or as an inquiry email the system reads and parses, with source recorded as the portal name (a **portal lead**). **The data contract:** each portal promises to deliver every inquiry it takes on the brokerage's listings, by whichever method that portal supports. The brokerage's system promises to capture every one of them, match it against existing leads and contacts by email or phone so a repeat inquiry is merged rather than duplicated (BR-6), and never quietly discard a lead just because the portal's format could not be parsed cleanly; that one still lands in Unrouted for a person to place (BR-7).

15. The agent notification channel: email and text message What crosses (out): the assigned agent's email and phone, and enough of the lead to act on it, the lead's name, phone, property of interest, and source, sent the moment a lead is assigned. **The data contract:** the notification channel promises immediate delivery once handed the message; the system promises to hand it over the instant an assignment happens, by both email and text together, never one without the other (BR-11). Because speed to first contact is the whole point of this flow, a delivery failure is not silent: the send is retried once, and if it still fails the Team Lead is alerted, the same escalation path used when a lead's first contact runs overdue (BR-12).

On confirming the specifics. The register above names the kind of connection each integration is (a modern MLS feed standard, a CRM, an IDX website, two named portals, email and text), which is enough to scope and build against. The exact regional MLS and its dialect of the feed standard, the specific CRM product in use, and the provider behind the notification channel are the details confirmed with the client before the build starts, and are carried as assumptions in [section 9](#).

4.9 Edge cases and exceptions

- **The MLS feed is down or slow:** the scheduled feed read does not complete. The sync run records the interruption as its outcome; already-synced listings keep their last known status, and no listing is marked off market on a failed run. The Listing Coordinator sees the failed or slow run on the Listings Sync Monitor and can trigger a Re-sync now once the feed is reachable again.
- **A listing is deleted from the feed:** it does not disappear from Harborstone's own listings. Per BR-14, a listing that leaves the feed is marked Off market, never deleted, so its history and any linked leads stay intact.
- **Two listings share a bad MLS number:** matching by MLS number (BR-2) cannot tell the two apart. The feed record is treated as a mapping problem (BR-4), and both affected listings are sent to Needs review on the Listing Review Queue, where the Listing Coordinator corrects the MLS number before either goes live.
- **A portal lead has no email:** the lead is still captured and normalised (lead capture). De-duplication (BR-6) falls back to phone; if neither email nor phone is present, the lead cannot be matched to anything and is treated as unreadable-source input (BR-7): captured as-is and sent to Unrouted for the Team Lead to review and assign by hand.
- **A lead comes in for a listing that is already sold:** the lead is still captured and routed normally (BR-8, BR-9); a lead references at most one property of interest (BR-15), and that property's current status (Sold or closed) shows alongside the lead so the assigned agent knows to manage expectations, not treat it as an active showing.
- **Every agent has Accepting-leads off:** BR-10 means the routing engine has no eligible agent to assign to, so the matched rule cannot complete and the lead falls to Unrouted (BR-9), the same as no rule matching at all. The Team Lead can still assign it manually from the Unrouted Leads screen, or turn an agent's Accepting-leads back on first.
- **A duplicate lead arrives days later:** BR-6 applies regardless of elapsed time. A new inbound record matching an existing lead's email or phone merges into that same lead (updating its source and message history) rather than creating a second lead, even if the original has already moved to Contacted, Working, or beyond.
- **The notification channel fails:** if the email or text send does not go through, the lead stays assigned (BR-11 does not make assignment conditional on delivery), and the assignment itself is unaffected. First-touch tracking (BR-12) is the backstop regardless of whether the notification reached the agent: if contact is not logged within the agreed time, it escalates to the Team Lead.
- **The feed sends a partial or garbled record:** treated the same as any other mapping or media problem (BR-4): the record is held at Needs review rather than published with missing or wrong information, and photos that are missing, too few, or the wrong format specifically trigger the same review (BR-5).

4.10 Engineering coverage: the build-blocking decisions

This is a new build assessed from a description, not an existing system inspected for problems, so this section sets out the **baseline the build implies** rather than findings against current code. The work it describes (the feed connection, the lead capture, the routing logic) is part of the §6 jobs; none of it is quoted here as a separate architecture or assessment line.

- **Integration data contracts.** Three inbound sources and one two-way link carry the whole automation: the MLS listing feed (read-only, via the RESO Web API), the listing portals Zillow and Realtor.com (inbound leads, by portal feed or parsed email), the brokerage's own IDX website (inbound inquiries), and the brokerage CRM (two-way: this system writes synced listings and leads into the CRM; the CRM's own updates to a lead's working status flow back in, so the agent's day-to-day notes in the CRM stay reflected here). Each inbound source authenticates with its own credential on file with that provider, and each is rate-limited by the provider's own terms, so the feed sync and the portal capture both read in bounded batches and back off rather than hammering the source. A read that fails is retried with a short back-off before it is treated as an error (see the concrete numbers below). (*Detailed per-integration contract: cross-reference §4.8.*)
- **Security & privacy.** A lead's contact details and message, and a listing's operational fields, are visible to the Listing Coordinator and the Team Lead; an Agent sees only the leads assigned to them and the listings they are the listing agent on (ties to §4.6). Only the Team Lead can change a routing rule or an agent's Accepting-leads flag (BR-13), so the routing behaviour itself cannot be altered by an agent. (See §8.)
- **Operational expectations.** The feed sync runs on a set schedule (not on demand for every visit), and every run is recorded as a sync run with its counts (seen, created, updated, flagged), visible to the Listing Coordinator on the Listings Sync Monitor with a manual Re-sync now option for a time- sensitive fix. Lead capture, by contrast, is treated close to real time: a portal lead or a website inquiry is picked up as it arrives, because speed to lead is the reason the routing and notification side of this automation exists at all. A run or a capture that fails outright is flagged for the Listing Coordinator rather than silently skipped, and the underlying listing and lead data is backed up on the same schedule as the rest of the brokerage's records. (*Cross-reference §8 for volume and availability.*)
- **Architecture-significant decisions.** This system is the source of truth for a listing's synced fields (fed one way from the MLS feed; never edited back into the feed) and for a lead's assignment (decided by the routing engine or, for an unrouted lead, by the Team Lead); the CRM is where the agent works the lead day to day, so only the lead's working status flows back from the CRM, never a competing assignment. This keeps the two-way link from producing two conflicting answers to "who is this lead assigned to." De-duplication is matched on the identifier each entity already carries: a listing by its MLS number (BR-2), a lead by email or phone (BR-6), so the same record is never entered twice regardless of which source it arrived from. The MLS feed side is schedule-based (poll), because the feed is a bulk export by nature; the lead-capture

side is arrival-based (each inquiry is handled as it lands), because a routed lead's value depends on how fast it reaches an agent. *(The exact split of what is authoritative where is confirmed with the client before build; see §9.2.)*

- **Build vs. buy.** The MLS-to-CRM connection itself is a solved problem in real estate: brokerages commonly buy or use a CRM's built-in feed connector rather than write their own feed reader, so this automation is expected to sit on top of a bought or CRM-native feed connection rather than parsing the RESO Web API from scratch. What is **built** is the part no packaged tool covers for this specific brokerage: the conflict-flagging review step, the lead de-duplication, the priority and round-robin routing rules, the Accepting-leads gate, and the first-touch escalation. Those rules are the brokerage's own policy, not a generic feature of a sync product, and they are the actual reason leads stop leaking and listings stop taking hours to re-key.

Concrete specifics (shown here, not deferred to a later audit). The MLS connection is drawn here as a standard authorised RESO Web API connection with a provider-set rate limit; on a failed read the feed sync retries with back-off (for example after 1, 5, and 30 minutes) before the run is marked with an error and left for the Listing Coordinator. The CRM link is shown as a standard two-way authorised connection to a mainstream real estate CRM; both are illustrative assumptions we would confirm before build (see §9.2). Listing and lead data is backed up daily and kept for the length of the brokerage's own record-keeping practice, then archived.

5. Acceptance criteria

Phase 1 is done when each of these holds. Each is something the brokerage can check for itself and a developer can test.

- **Listings arrive automatically.** A new listing in the MLS appears in the CRM and on the website within the agreed sync window, with its photos and its status matching the MLS. How to check: add or change a test listing in the MLS, wait one sync window, confirm it matches on both.
- **Bad listings are held, not published.** A listing with a mapping or media problem appears in the Listing Review Queue and does not go live until a coordinator resolves it. How to check: feed a listing with a broken photo, confirm it lands in the queue and not on the site.
- **Status stays true to the feed.** A listing that goes under contract, closes, or leaves the MLS shows the matching status, and one that leaves the feed becomes Off market rather than disappearing. How to check: change a test listing's status in the MLS and confirm the CRM follows.
- **Leads arrive in seconds, de-duplicated.** A website or portal inquiry appears in the CRM within seconds, and a second inquiry from the same person is merged, not duplicated. How to check: submit a test inquiry twice, confirm one lead with both touches.

- **Leads route and notify.** A new lead is assigned to an agent by the routing rules and that agent is notified by email and text; a lead no rule matches lands in Unrouted for the Team Lead. How to check: submit leads that match and do not match a rule, confirm the assignment, the notification, and the Unrouted fallback.
- **Only the right people, only available agents.** Only the Team Lead can edit routing rules and agent availability, and only an agent who is accepting leads receives one. How to check: turn an agent's availability off and confirm they stop receiving leads.
- **First contact is measured.** First contact on a lead is timestamped, and a lead not contacted within the agreed time escalates to the Team Lead. How to check: leave a test lead untouched past the limit and confirm the escalation.

6. Phased build plan

Phase 1 (the slice that pays for itself) is the top of the §3 opportunity ranking: sync listings from the MLS automatically, capture and route leads by rule within seconds, and give each role the screens to watch it. That is the slice that removes the two manual data-entry jobs and fixes the speed-to-lead problem, and it is sized to the Connected Systems tier (a floor of 100 developer-hours; this Phase 1 is 170 hours, above the floor). The work is grouped into **three epics**, by significance; under each epic the **jobs** say what is built, why it matters, and how it is achieved; and under each job the **steps** are the actual build, **each one small**, no step longer than two days, so you see exactly what is being done and what it costs. The epic and job figures are the sum of their steps. The delivery work each job implies (its screen design, its tests, the shared setup it rides on) is folded into the steps, so the total is the whole shippable build with nothing left out. Duration is in working days; a developer does 4 to 6 productive hours in a day.

WORK AND STEPS	DURATION	COST
Epic 1. Listings sync	~12-17 d	\$7,000
MLS feed sync engine. The automation at the heart of the listings flow: on a schedule it reads the MLS feed and creates or updates each listing, so the coordinator stops re-keying and the site stays true to the MLS. <i>How we achieve it: read the feed on a schedule, map its fields to the listing, match by MLS number with a dedupe guard, handle photos, drive the status changes from the feed, and flag what it cannot place cleanly.</i>	~7.5-11 d	\$4,400
Read the MLS feed on a schedule Read the feed through the connection in §4.8 on a schedule, with retry and rate handling so a slow feed never loses a run.	1.5-2 d	\$1,000
Map fields and match by MLS number Map the feed fields to the Listing (§4.1) and match by MLS number, creating a new listing or updating the existing one (BR-2).	1.5-2 d	\$900

WORK AND STEPS	DURATION	COST
De-duplicate and handle photos Guard against duplicates and bring in the listing photos and media cleanly.	1.5-2 d	\$800
Drive listing status from the feed Move each listing between Active, Under contract, Sold and Off market as the feed changes (BR-1, BR-3, BR-14, §4.5).	1-1.5 d	\$700
Flag what cannot be placed to review Send a listing with a mapping or media problem to Needs review instead of live (BR-4, BR-5, §4.9).	1-1.5 d	\$600
Record each sync run, safe to re-run Record each run (seen, changed, errors) and make a re-run safe with no duplicates (§4.1).	0.5-1 d	\$400
Sync monitor and review queue. The coordinator's two screens: the monitor that shows the feed is healthy, and the queue of the few listings that need a human look. It replaces watching the MLS by hand with a short worklist. <i>How we achieve it: build the monitor and the review queue with the resolve and re-sync actions, and their empty and error states.</i>	~4.5-6.5 d	\$2,600
Listings Sync Monitor screen The Listings Sync Monitor (§4.2): last run, counts and errors, with a Re-sync now button.	1.5-2 d	\$800
Listing Review Queue with a badge The Listing Review Queue (§4.2) of listings needing a look, with a badge and the resolve action.	1.5-2 d	\$800
Listing Detail and manual re-sync The Listing Detail screen (§4.2) with sync status, and a manual re-sync trigger.	1-1.5 d	\$600
Empty and error states The empty and error states across the two screens so the coordinator always knows where things stand.	0.5-1 d	\$400
Epic 2. Lead capture and routing	~14-19 d	\$7,600
Lead capture from the website and portals. Bring every inbound lead into the CRM the moment it arrives, de-duplicated, so none sit in an inbox and none are entered twice. <i>How we achieve it: capture leads from the IDX site and the portals, normalise them to the lead, and de-duplicate on the way in.</i>	~5-7.5 d	\$3,000
Capture website inquiries Capture inquiries from the brokerage IDX website as leads (the connection in §4.8).	1.5-2 d	\$800
Capture portal leads (Zillow, Realtor.com) Capture leads from the listing portals through their feed or parsed email (§4.8).	1.5-2 d	\$1,000

WORK AND STEPS	DURATION	COST
Capture an unreadable source as-is Capture a source that cannot be read automatically as-is and send it to Unrouted (BR-7, §4.9).	0.5-1 d	\$400
Routing engine and agent notification. The automation that gets each lead to an agent in seconds and tells them at once. This is what turns a 2-to-4-hour first touch into minutes and stops the leak. <i>How we achieve it: evaluate the routing rules, assign or fall to Unrouted, respect availability, and notify the agent instantly with a first-touch timer.</i>	~4-6 d	\$2,400
Evaluate routing rules by priority Evaluate the rules by priority and round-robin within a rule to share the load (BR-8).	1.5-2 d	\$800
Fall to Unrouted, respect availability Fall to Unrouted when nothing matches, and only route to an agent who is accepting leads (BR-9, BR-10).	1-1.5 d	\$600
Notify the agent by email and text Notify the assigned agent by email and text the moment a lead lands (BR-11, §4.8).	1-1.5 d	\$600
First-touch timer and escalation Stamp first contact and escalate an overdue lead to the Team Lead (BR-12).	1-1.5 d	\$400
Routing-rule screen and lead inbox. The screens the Team Lead and agents use around the routing: the rule editor, the roster, and the agent's own inbox. <i>How we achieve it: build the Team Lead's rule editor and roster and the agent's lead inbox.</i>	~4-5.5 d	\$2,200
Routing Rules list and editor The Routing Rules screen (§4.2): match on, assign to, priority and active, editable by the Team Lead only (BR-13).	1.5-2 d	\$800
Agent Roster and availability The Agent Roster screen (§4.2) with the Accepting-leads toggle.	1-1.5 d	\$600
Agent Lead Inbox with first-touch timer The agent Lead Inbox (§4.2): assigned leads, status, and the first-touch timer.	1.5-2 d	\$800
Epic 3. Data import and go-live	~4-6.5 d	\$2,400
Data import and go-live. The shared work that serves every screen above and cannot fold into one of them: bring the brokerage's roster, open leads and active listings in, then deploy and hand over. <i>How we achieve it: import each dataset behind a dry-run report and reconcile it, then turn the §5 checks into tests, deploy, and walk each role through their screen once.</i>	~4-6.5 d	\$2,400
Import the roster and open leads Import the agent roster and existing open leads behind a dry-run report, reconciling before go-live (§7).	1-2 d	\$700

WORK AND STEPS	DURATION	COST
Import active listings and reconcile Import the active listings and reconcile the count against the MLS and the CRM (§7).	1-1.5 d	\$600
Wire the acceptance checks as tests Turn the §5 acceptance checks into automated tests that run on every change.	1-1.5 d	\$600
Go-live deploy and walkthrough Deploy to production and walk each role through their screen once so they own it on day one.	1-1.5 d	\$500
Total for Phase 1	~28-42 d	\$17,000

The three epics above are the whole Phase-1 build: every job is broken into named steps, each priced and **none longer than two days**, so the plan is visible all the way down rather than hidden inside a job total. There is no separate engineering or architecture line: the integration handling, the sync schedule and the access rules are built into the steps that need them, not a phase of their own. Anything that would push past the slice (the two-way deal sync, the reporting) stays in the Phase 2 sketch below. The scope fits the Connected Systems ceilings (3 roles, well under 15 steps, 5 integrations, 5 kinds of information, 7 screens, 15 rules, branching lifecycles), and is specified to build with no further discovery.

Phase 2 (sketch): close the loop back to the deal, and turn the data into insight. Once listings and leads flow reliably, the next phase adds the pieces that depend on a tool choice or on Phase 1 data existing first. *What is suggested.* A two-way sync of deal and transaction status between the CRM and the brokerage's transaction-management tool, so a listing moving to under contract and closing is reflected on both sides without anyone re-keying it; and reporting that shows which lead sources return the most closed business and how fast each agent makes first contact. *How the system benefits.* It gives the brokerage one true pipeline across its tools, and it turns the captured lead and listing data into decisions about where to spend and how the team is performing. *Why this is a later phase.* The two-way deal sync depends on the brokerage choosing its transaction tool (the dependency in §7), and the reporting earns its keep only once Phase 1 has been running long enough to have data worth reading. It is sketched, not quoted: expect another Connected Systems-scale slice, shaped once Phase 1 is live.

7. Risks and dependencies

- **MLS feed access and terms.** The build depends on the brokerage's MLS granting feed access (an IDX or RESO Web API feed) and on its rules for how listings may be displayed. If access is slow to grant or the terms limit what can be shown, the listings flow waits on it. This is the main external dependency, worth starting first.

- **Portal lead delivery varies.** Zillow and Realtor.com deliver leads differently (a direct feed for some accounts, parsed email for others). The exact method per portal is confirmed at the start so the capture is built to the real source.
- **Feed data quality.** Listing feeds carry the odd malformed or incomplete record. The review queue handles these, but a feed with many bad records means more coordinator review at first; it settles as the mapping is tuned.
- **Notification deliverability.** Email and text alerts depend on a sending provider and on agents not filtering them out. A short setup and a test to each agent removes most of this.
- **The transaction tool is a Phase 2 dependency.** The two-way deal sync sketched in §6 depends on the brokerage choosing its transaction-management tool. Until then that piece stays out of scope.
- **Existing data to import.** Go-live depends on the brokerage providing its current agent roster, open leads and active listings in a readable form; the dry-run import surfaces problems before they matter.

8. Non-functional expectations

- **Speed.** A lead is captured and routed within seconds of arriving, so an agent can make first contact in minutes. Listings sync on a schedule agreed with the brokerage, often several times a day, so the site is never far behind the MLS.
- **Availability.** The lead capture and notification run around the clock, because leads arrive at all hours and a missed night is lost business. If the site or a portal is briefly unreachable, captured leads are held and processed when it returns, none are dropped.
- **When something fails.** A failed sync run is retried and, if it keeps failing, raised on the Sync Monitor so the coordinator sees it the same hour. A failed notification is retried and the lead still appears in the agent's inbox, so a delivery problem never loses the lead itself.
- **Backups and recovery.** The listing and lead data is backed up regularly so the brokerage's records can be restored. Because listings can be re-synced from the MLS, a listing gap is recoverable from the feed as well.
- **Security and privacy.** Lead contact details are personal data: they are visible only to the Team Lead, the assigned agent, and the brokerage's own administrators, and access follows the role rules in §4.6. Feed and CRM connections use the providers' secure, authorised access, not shared logins.
- **Growth.** The volumes here (about 200 leads a month, dozens of listing changes a week, about 20 agents) are comfortably within a normal build; the same system carries a larger brokerage without a redesign.

9. What still needs confirming

These are the points still open before the build is a green light. None of them have been guessed into the specification above; the spec stays grounded and the uncertainty lives here, named, in three groups.

9.1 Open questions

1. How often should listings sync from the MLS: several times a day, hourly, or as near to real time as the feed allows?
2. What first-contact time should a lead have before it escalates to the Team Lead?
3. Which routing model does the brokerage want to start with: round-robin across the team, by territory, by lead source, or by price band?
4. Should buyer leads and seller leads route differently, or through the same rules?

9.2 Assumptions we made (to confirm)

Where the brief was silent we proceeded on a reasonable assumption rather than leaving a hole, so the picture is complete. Each is shown for confirmation, not asserted as fact.

1. The cost of the status quo in §3 rests on these figures: a loaded coordinator rate of about \$35 an hour, about 8 hours a week of listing re-keying, about 200 inbound leads a month, and an average closed-transaction value of about \$9,000. They size the roughly \$4,000 to \$7,000 a month figure.
2. The specific systems are named illustratively: the MLS read through a RESO Web API feed, the brokerage CRM, the portals Zillow and Realtor.com, and an email and text notification provider. The real feed dialect, CRM product, portal set, and provider are confirmed at the start. The data contracts in §4.8 stay the same; only the connectors change.
3. We assumed the MLS number is the regional MLS-issued identifier and is unique per listing, which is what the create-or-update match in §4 relies on.
4. We assumed the system owns lead assignment while the CRM stays the source of working status, so the two systems do not fight over the same field. This split is confirmed before build.
5. The new system introduces a few things the manual way does not have, shown here so the brokerage can confirm them: a **sync run** record kept for each poll of the feed (what it saw, changed, and flagged) with a **last-synced** time and a **sync status** on each listing; a **first-contact** timestamp on each lead; an **accepting-leads** switch on each agent; and the **routing rule** set itself (its match, its assignment, its priority, and whether it is active). These are design points, not facts pulled from the current process.
6. The first-contact window shown on the Lead Inbox is illustrative; the real target is set in question 2.

7. We assumed a retry schedule of about 1, 5, and 30 minutes on a failed sync run or notification before it is raised for attention.

9.3 Requirements before we start

What the brokerage needs to have in place before Phase-1 build begins, so neither side is held up on day one.

1. MLS feed access and the display terms that go with it, so the listings sync can be set up and tested.
2. CRM access that allows listings and leads to be written in and assignment read back.
3. Confirmation of how each portal delivers its leads (a direct feed or a parsed email), per portal.
4. The current agent roster, open leads, and active listings, exported for the one-time import in §6.
5. Sign-off on the routing rules, the first-contact window, and the wording of the agent notifications.
6. A short window of the Team Lead's and the coordinator's time to walk through their screens once ready, and to confirm the first routed leads and synced listings before the team relies on them.