

17 OCTOBER, 2025



Northstar Residential

NORTHSTAR RESIDENTIAL SALESFORCE TECHNICAL FINDINGS & PLATFORM ANALYSIS

PREPARED BY CLEARWORK STUDIOS
FICTIONALIZED PORTFOLIO SAMPLE

Table of Contents

0 DOCUMENT CONTROL2

0.1 DOCUMENT INFORMATION2

0.2 REVISION HISTORY2

0.3 FICTIONALIZED NOTICE.....2

1 DOCUMENT OVERVIEW.....3

1.1 PURPOSE3

1.2 INTENDED AUDIENCE3

1.3 ASSESSMENT SCOPE.....3

2 FINDINGS AT A GLANCE4

2.1 PRIMARY THEMES4

3 APEX & AUTOMATION FINDINGS.....5

3.1 REVIEWED COMPONENT SUMMARY5

3.2 REPRESENTATIVE CODE FINDINGS.....5

3.3 AUTOMATION FINDINGS5

3.4 RECOMMENDED DIRECTION6

4 LEAD INTAKE & ASSIGNMENT6

4.1 CURRENT-STATE PROCESS6

4.2 OBSERVED CONDITIONS6

4.3 PROPOSED FUTURE-STATE FLOW.....7

4.4 RECOMMENDATIONS.....7

4.5 SUGGESTED METRICS7

5 INTEGRATION ARCHITECTURE8

5.1 SYSTEM LANDSCAPE8

5.2 OBSERVED ISSUES & GAPS9

5.3 GOVERNANCE OPPORTUNITIES.....9

6 UNIFIED INTEGRATION MONITORING FRAMEWORK9

6.1 PURPOSE9

6.2 CORE COMPONENTS10

6.3 OPERATIONAL ACTIONS10

6.4 SUGGESTED METRICS10

6.5 ALERTING.....11

7 DATA MODEL & GOVERNANCE11

7.1 REPRESENTATIVE OBJECT MODEL.....11

7.2 OBSERVATIONS13

7.3 RECOMMENDED GOVERNANCE PRACTICES13

8 DEPLOYMENT FRAMEWORK.....13

8.1 RECOMMENDED ENVIRONMENT PATH13

8.2 REQUIRED DELIVERY CONTROLS.....14

9 PRIORITIZED NEXT STEPS.....15

10 CLOSING SUMMARY15



0 DOCUMENT CONTROL

0.1 DOCUMENT INFORMATION

Document Name	Northstar Residential Salesforce Technical Findings & Platform Analysis
Document Version	1.1
Publication Date	10/17/2025
Prepared For	Northstar Residential
Prepared By	Clearwork Studios
Classification	Representative Fictionalized Portfolio Sample

0.2 REVISION HISTORY

Version	Published	Summary	Initials
1.0	10/15/2025	Initial Publication	KE
1.1	10/17/2025	Updated terminology; corrected minor formatting inconsistencies	KE

0.3 FICTIONALIZED NOTICE

This document is a representative portfolio sample reconstructed from prior consulting experience. Northstar Residential is a fictional organization. All organization names, systems, metadata, component names, technical findings, quantities, diagrams, thresholds, and recommendations were newly created or materially altered to protect confidential client information.

1 DOCUMENT OVERVIEW

1.1 PURPOSE

This document summarizes technical findings from a focused review of Northstar Residential's Salesforce platform and selected connected systems.

The assessment examined the platform's current automation, custom code, lead-intake process, integration architecture, data model, and deployment practices. Its purpose is to give Northstar's Salesforce and IT teams a shared technical reference for validating current-state conditions, discussing risk, and prioritizing future remediation.

This document is not intended to serve as a complete modernization roadmap. Recommendations are presented at a directional level to support technical planning and follow-up discovery.

1.2 INTENDED AUDIENCE

This report is intended for:

- Salesforce administrators and developers responsible for maintaining the platform
- IT and integration teams supporting connected systems
- Technical and operational leaders evaluating platform risk and investment priorities
- Delivery teams responsible for planning remediation and modernization work

1.3 ASSESSMENT SCOPE

1.3.1 In Scope

- Apex classes and triggers
- Salesforce Flow and legacy automation
- Lead intake, matching, and assignment
- Integration architecture and operational monitoring
- Selected custom objects and relationships
- Environment and deployment practices

1.3.2 Out of Scope

- Full source-code review
- Penetration or security testing
- Business-process redesign
- End-user adoption assessment
- Detailed implementation estimates
- Vendor-contract or licensing review

1.3.3 Assessment Basis

Findings are based on a representative review of a production-aligned test environment, technical interviews, metadata inventories, architecture materials, and selected operational examples.

2 FINDINGS AT A GLANCE

Northstar's Salesforce platform supports core sales, buyer, community, and integration processes. The environment is operational, but several areas would benefit from stronger consistency, ownership, and technical governance.

Area	Current-State Finding	Risk	Recommended Direction
Apex and triggers	Several classes combine orchestration, validation, and data access	High	Separate responsibilities and introduce shared service patterns
Salesforce automation	Multiple flows update the same objects with inconsistent fault handling	High	Consolidate orchestration and standardize error paths
Lead intake	Matching occurs after some downstream processing has begun	Medium	Normalize and match leads earlier in the transaction
Integrations	Salesforce errors are visible, but external queue health is not	High	Introduce unified cross-system monitoring
Data model	Similar lifecycle fields exist across several objects	Medium	Establish object ownership and naming standards
Deployment	Release validation relies heavily on team knowledge and manual steps	Medium	Formalize promotion gates and deployment evidence

2.1 PRIMARY THEMES

Visibility: Technical teams can identify many Salesforce-side failures, but they do not have one place to view the full transaction lifecycle across Salesforce and connected platforms.

Consistency: Code, flows, metadata, and deployment practices have evolved over time. Equivalent functions are not always implemented using the same patterns.

Ownership: Responsibility for integrations, data definitions, exception queues, and technical documentation is not consistently assigned.

Maintainability: The platform functions under normal conditions but troubleshooting and change implementation often require specialized institutional knowledge.

Overall Assessment: The environment does not require wholesale replacement. The strongest path forward is targeted modernization supported by clearer ownership, standardized patterns, and stronger operational monitoring.

3 APEX & AUTOMATION FINDINGS

3.1 REVIEWED COMPONENT SUMMARY

The fictionalized assessment identified:

Component Type	Reviewed	Active or Relevant
Apex classes	62	48
Apex triggers	9	7
Record-triggered flows	34	26
Scheduled or auto-launched flows	11	8
Managed-package automations	14	14

3.2 REPRESENTATIVE CODE FINDINGS

Component	Function	Observation	Priority
LeadAssignmentService	Matches and routes incoming leads	Combines matching, assignment, notifications, and logging in one service	High
BuyerLifecycleTrigger	Updates buyer milestone fields	Performs repeated queries during record processing	High
CommunitySyncHandler	Prepares community records for integration	Error responses are logged inconsistently	Medium
WarrantyPackageAdapter	Supports managed-package activity	Direct changes require coordination with the package provider	Monitor

3.3 AUTOMATION FINDINGS

The reviewed automations showed several recurring patterns:

- Multiple flows update the same Lead and Opportunity fields.
- Fault connectors are not applied consistently.
- Some automation names describe implementation history rather than business purpose.
- Ownership is not documented for several business-critical flows.
- Managed-package components cannot be remediated using the same approach as custom automation.
- Several flows contain notification or field-update logic that could be consolidated.

3.4 RECOMMENDED DIRECTION

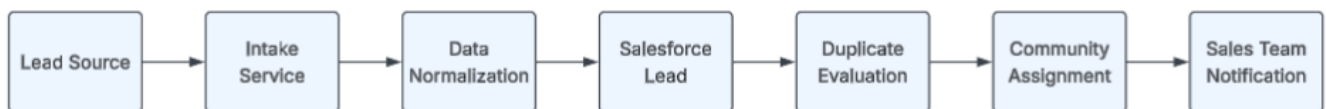
1. Establish one orchestration flow per major object and business event.
2. Move reusable decision logic into subflows or shared Apex services.
3. Require fault handling for every flow that performs record creation, external calls, or multi-step updates.
4. Assign a business owner and technical owner to each critical automation.
5. Apply a consistent naming structure that identifies object, trigger, purpose, and execution timing.
6. Review custom and managed-package automation separately so recommendations account for vendor constraints.

4 LEAD INTAKE & ASSIGNMENT

4.1 CURRENT-STATE PROCESS

Northstar receives leads from its public website, community landing pages, referral partners, and third-party listing services.

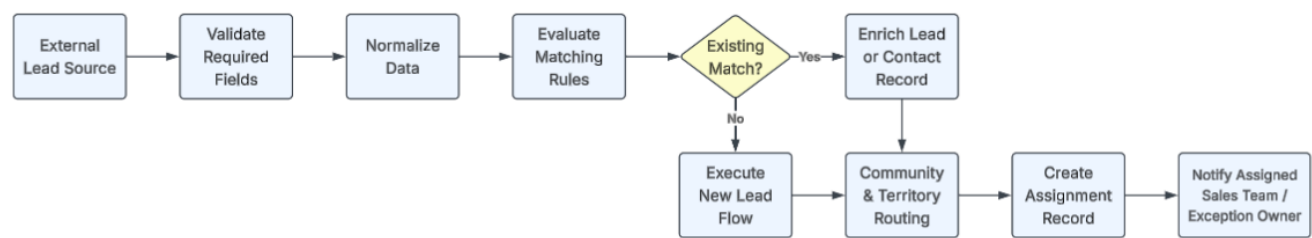
A typical lead follows this path:



4.2 OBSERVED CONDITIONS

- Email addresses and phone numbers are not normalized consistently before matching.
- Duplicate checks rely heavily on exact values.
- Some assignment rules are stored in code while others are maintained in Flow.
- Incomplete community or region data can prevent automatic routing.
- Failed records are logged, but not every failure receives an operational owner.
- Intake metrics focus on total volume rather than duplicate rate, routing failures, or resolution time.

4.3 PROPOSED FUTURE-STATE FLOW



4.4 RECOMMENDATIONS

Recommendation	Intended Outcome
Move matching configuration into custom metadata	Allows administrators to revise rules without code deployments
Normalize source data before matching	Improves duplicate detection
Create a dedicated exception queue	Gives unresolved leads clear ownership
Track source-level quality metrics	Identifies unreliable lead providers
Record the matching rule used	Improves auditability and troubleshooting
Establish resolution targets	Prevents failed leads from remaining unassigned

4.5 SUGGESTED METRICS

- Lead acceptance rate
- Duplicate rate by source
- Automatic assignment rate
- Exception-queue volume
- Average exception resolution time
- Leads created without a community or region
- Repeat inquiries connected to an existing buyer

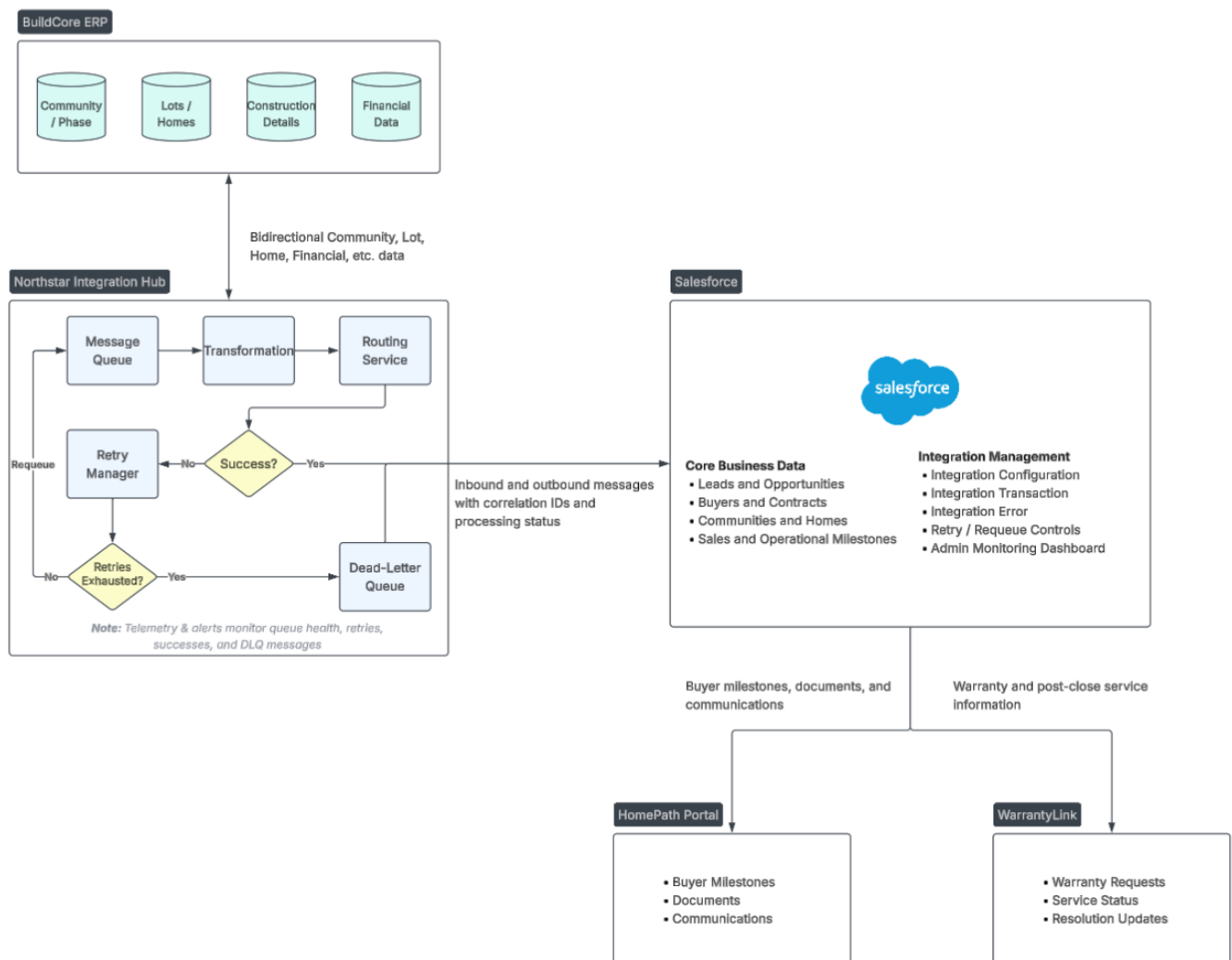


5 INTEGRATION ARCHITECTURE

5.1 SYSTEM LANDSCAPE

Northstar Residential's Salesforce platform sits at the center of a broader ecosystem supporting sales, construction, buyer communication, and post-close service. Data moves between Salesforce, the BuildCore ERP, and downstream customer-facing applications through the Northstar Integration Hub.

The following diagram illustrates the current system landscape, including the primary business data exchanged between platforms and the integration components used for transformation, routing, retry handling, exception management, and operational monitoring.



5.2 OBSERVED ISSUES & GAPS

Category	Observation	Operational Impact
Retry logic	All failures use the same retry limit	Transient and permanent errors are treated alike
Monitoring	Salesforce does not display external queue status	Problems may surface first as data discrepancies
Throughput	Some message types are processed individually	High volume creates unnecessary queue traffic
Logging	Not every exception produces a structured error record	Trend and root-cause analysis remain incomplete
Configuration	Several endpoints and thresholds are code-managed	Routine changes require technical deployment
Ownership	External-system support responsibility varies	Resolution can stall between teams

5.3 GOVERNANCE OPPORTUNITIES

- Maintain a central integration registry.
- Assign a business and technical owner to every interface.
- Require contract and schema review before integration changes.
- Define standard retry categories and escalation paths.
- Align retention and privacy rules across all connected platforms.
- Use shared integration patterns for future systems.

6 UNIFIED INTEGRATION MONITORING FRAMEWORK

6.1 PURPOSE

Northstar's current integration design provides reliable asynchronous processing under normal conditions, but operational visibility is divided across Salesforce and the Integration Hub.

The proposed framework creates a single operational view for identifying failures, reviewing transaction history, managing retries, and escalating unresolved incidents.

6.2 CORE COMPONENTS

Component	Purpose
Integration Registry	Stores interface owners, endpoints, message types, thresholds, and support information
Transaction History	Records each processing attempt, duration, outcome, and correlation ID
Retry Manager	Applies configurable retry rules based on error category
Exception Queue	Holds transactions requiring review or corrected data
Salesforce Admin Dashboard	Displays transaction health and provides business-context visibility
Integration Hub Telemetry	Tracks queue depth, latency, dead-letter volume, and service health
Incident Record	Coordinates high-severity failures through assignment, status, and resolution notes

6.3 OPERATIONAL ACTIONS

Authorized administrators should be able to:

- review the complete history of a transaction
- identify the failing stage
- distinguish transient from data-related errors
- correct eligible business data
- retry or requeue a transaction
- assign an incident to the correct owner
- confirm final resolution
- analyze recurring failure patterns

6.4 SUGGESTED METRICS

Metric	Purpose	Example Threshold
Transaction success rate	Measures daily integration stability	At least 97.5%
Average retry count	Identifies unstable services	Fewer than 3 attempts
Dead-letter queue depth	Tracks unresolved failures	Fewer than 20 open
End-to-end processing time	Measures integration latency	Under 8 seconds
Critical error resolution	Measures support response	Under 4 business hours
Repeat failure rate	Identifies unresolved root causes	Under 5%

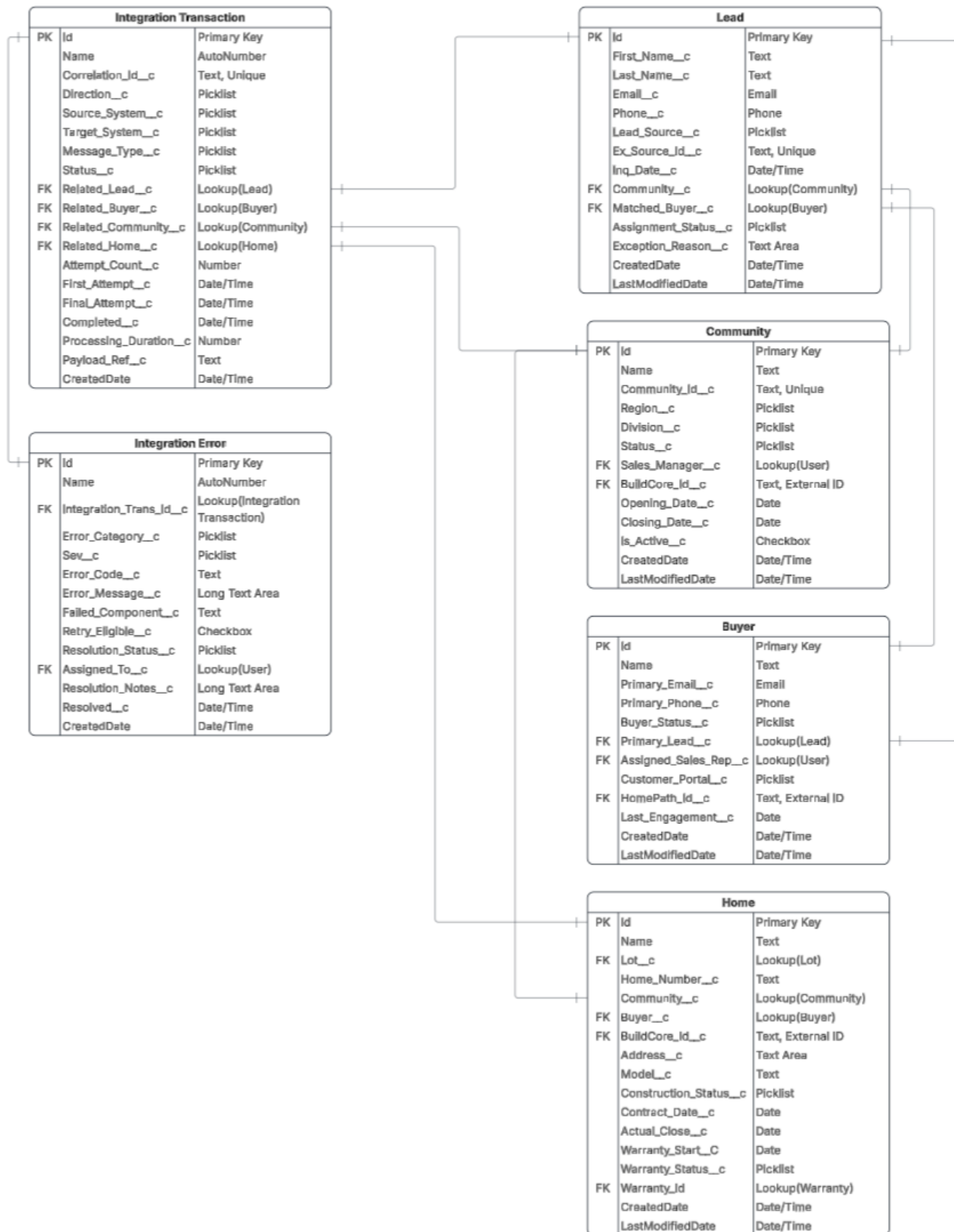
6.5 ALERTING

- Warning alerts for growing queue depth or elevated retries
- Critical alerts for unavailable endpoints or dead-letter accumulation
- Automatic incident creation for high-severity conditions
- Email or collaboration-platform notifications based on severity
- Escalation when an incident exceeds its target resolution time

7 DATA MODEL & GOVERNANCE

7.1 REPRESENTATIVE OBJECT MODEL

The following diagram presents a representative subset of Northstar Residential's Salesforce data model. Core objects are shown in greater detail, while supporting entities such as Lot and Warranty are included only to clarify key relationships. Additional standard objects, junction objects, historical records, and lower-level technical relationships have been omitted for readability.



Representative model only. Not all platform objects, fields, or relationships are shown.

7.2 OBSERVATIONS

- Similar status fields are maintained on Leads, Buyers, and Homes.
- Some fields describe temporary process state but remain indefinitely.
- Object and field descriptions are incomplete.
- Ownership is unclear for several legacy fields.
- Integrations write directly to business objects without a consistent staging pattern.
- Naming conventions differ between older and newer components.

7.3 RECOMMENDED GOVERNANCE PRACTICES

Ownership: Assign one business owner and one technical owner to each core object.

Data Dictionary: Document field purpose, source, allowed values, downstream use, and retention requirements.

Lifecycle Standards: Define standard terminology for inquiry, qualification, contract, construction, close, and warranty stages.

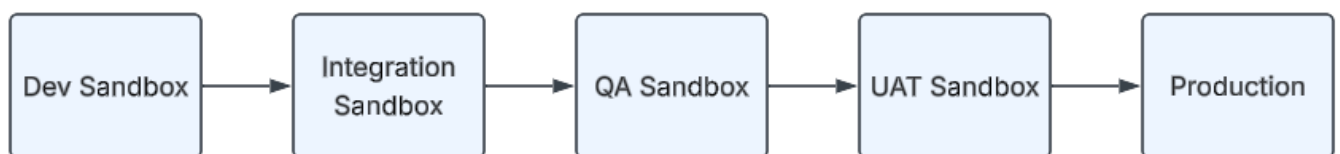
Integration Boundaries: Identify which platform owns each data element and when Salesforce may update it.

Change Review: Require impact analysis for new fields, status changes, required values, and integration mappings.

Routine Cleanup: Review unused fields, duplicate values, inactive automation, and stale integrations quarterly.

8 DEPLOYMENT FRAMEWORK

8.1 RECOMMENDED ENVIRONMENT PATH



8.2 REQUIRED DELIVERY CONTROLS

8.2.1 Development

- Link each change to an approved work item.
- Create changes in an isolated development environment.
- Document configuration and post-deployment requirements.
- Complete peer review before promotion.

8.2.2 Integration Validation

- Confirm related automation and interfaces operate together.
- Run automated validation and regression tests.
- Verify integration configuration for the target environment.
- Record defects and remediation decisions.

8.2.3 Quality Assurance Testing

- Deploy the complete release candidate and dependencies.
- Validate acceptance criteria and exception paths.
- Perform targeted regression and integration testing.
- Record evidence, defects, and retest results.
- Resolve release-blocking issues before UAT.
- Document and approve any deferred defects.

8.2.4 User Acceptance Testing

- Validate acceptance criteria with business stakeholders.
- Document test evidence and approval.
- Resolve or formally defer identified issues.
- Confirm deployment timing and communication needs.

8.2.5 Production Release

- Run pre-deployment validation.
- Capture a rollback or remediation plan.
- Complete required manual steps.
- Perform post-deployment smoke testing.
- Record deployment evidence and final approval.

9 PRIORITIZED NEXT STEPS

9.1 FIRST 30 DAYS

- Assign owners to critical integrations and core objects.
- Centralize the component and interface inventories.
- Define severity, priority, and escalation standards.
- Identify the highest-risk automation overlaps.

9.2 DAYS 31–60

- Pilot unified monitoring for one outbound integration.
- Introduce standard flow fault handling.
- Document the environment and deployment process.
- Create a technical-governance review cadence.

9.3 DAYS 61–90

- Begin remediation of the highest-priority code and automation.
- Implement the integration registry.
- Establish schema and metadata governance.
- Measure operational performance against the new monitoring thresholds.

10 CLOSING SUMMARY

Northstar's Salesforce platform provides a stable foundation for sales and operational coordination. Its next stage of maturity depends less on introducing new technology than on creating clearer ownership, stronger visibility, and repeatable technical standards.

Targeted modernization of automation, lead intake, integration monitoring, data governance, and release management will reduce reliance on institutional knowledge while improving the platform's resilience and maintainability.