

Reducing Duplicate Identity Creation in Enterprise CRM Through Guided Intake Workflows: A Cloud and Digital Experience Platform Approach

Ravi Chandra Chinta,

Independent Researcher

Abstract: A duplicate borrower record is rarely a one-time mistake with a one-time cost. It propagates - a second Contact, a second Account, sometimes a second portal login - until two separate histories exist for what should be one relationship. This article argues that the standard fix for this problem, algorithmic deduplication applied after records already exist, treats a symptom that guided intake design can prevent at the source. The evidence comes from an architectural case: a guided intake workflow built for retail and brokered loan origination on a cloud CRM platform with an integrated Digital Experience Platform. The redesign enforces a search-before-create sequence, requires explicit borrower-and-entity linkage decisions, and routes ownership conflicts through a transfer request rather than allowing a second record to appear quietly. It also extends into the integration layer, replacing a manual file upload with an authenticated API contract between the originating system and the CRM. Loan-creation time and duplicate-record rates both improved substantially in this implementation; the more interesting claim, though, is architectural rather than numeric. Prevention-oriented intake design is underdeveloped relative to the entity-resolution literature it complements - a gap this article maps and then closes with a transferable design pattern for architects working on customer intake within Digital Experience Platforms.

Keywords: NA

1. Introduction

A loan officer who creates a duplicate borrower record has not made an isolated clerical error. That duplicate becomes a second Contact, potentially a second Account, and - once a customer-facing portal enters the picture - possibly a second login tied to the wrong history of the relationship. Cloud CRM platforms are built to make record creation fast, which is exactly what makes them useful for sales teams under pressure. The same ease of creation, though, is what lets duplicate identities enter a system with almost no resistance.

Most of the response to this problem, in both industry practice and the research literature, arrives after the fact. Entity resolution, record linkage, and blocking algorithms find and merge duplicates once they already exist in the database [1], [2], [3], and the technical sophistication of that work has grown considerably - recent contributions have pushed scalability and matching accuracy well past what heuristic approaches could manage [4], [5], [6]. None of this is wrong. It is, however, incomplete on its own terms: it treats duplication as inevitable residue to be cleaned rather than as a defect that a business process could simply avoid producing. Where record creation happens through a defined workflow - a sales intake screen, a service request, an onboarding form - prevention at the point of intake is not just feasible; it is underused as a design lever, and the literature on workflow design as a data-quality control has not kept pace with the literature on cleaning up after the fact.



The case behind this argument is a guided “Start Loan” workflow on a cloud CRM platform (Salesforce Lightning), paired with a customer-facing Digital Experience Platform (Experience Cloud), supporting retail and brokered loan origination. The prior process used standard lead conversion with no enforced search step. Training did not fix the resulting duplicate rate - moving fast paid better, in practice, than searching first did. The redesign removes that tradeoff by making the correct sequence the only sequence available: search before create, explicit entity decisions, ownership-aware routing, and a confirmation step before anything commits.

Three bodies of work rarely appear together in a single design argument: entity-resolution theory, multidimensional data-quality frameworks, and Digital Experience Platform architecture. This article brings them together into one implementable intake pattern, then extends that pattern into an integration design that removes the last manual step - a file export and upload - from the pipeline entirely. What follows reviews the relevant literature, sets out the analytical method, describes the workflow architecture and its integration extension, reports case-based outcomes, and discusses where the pattern travels well and where it does not.

2. Literature Review

2.1 Entity resolution's post-hoc bias

Entity resolution is a mature field. Fellegi and Sunter's probabilistic framework [1] still anchors the discipline nearly sixty years after its publication, and contemporary surveys trace how far matching techniques have advanced toward scalable, learned approaches [2], [3]. Recent empirical work extends the trajectory further: deep-learning blocking methods now outperform hand-tuned heuristics on standard benchmarks [4], large-scale filtering benchmarks continue refining what “efficient” means at enterprise volumes [5], and machine-learning deduplication pipelines validated against reference datasets push the work toward production readiness [6].

One assumption unites nearly all of it: duplicates already exist, and the task is to find them. Binette and Steorts frame entity resolution explicitly as inference over data already collected [7]. That framing is defensible on its own terms, but it quietly closes off a different question - whether the rate at which duplicates enter a system in the first place is itself something a designer can control. A CRM instance that produces fewer duplicates at intake reduces the load on every downstream matching pipeline. Intake-workflow design, though, rarely shows up as a variable in this literature at all.

2.2 Where guided workflow belongs as a quality control

Wang and Strong's multidimensional data-quality framework [8] treats accuracy, completeness, and consistency as measurable properties - a framing more recent data-quality scholarship continues to extend [9], [10]. What these frameworks tend to share is a habit of treating “process” as an antecedent to be measured after the fact rather than something to be engineered directly. Breitmayer and colleagues break from that pattern: their empirical study of user editing behavior in electronic forms shows that guided, constrained form design measurably changes how people enter data [11]. This article pushes that finding one level up - from constraint at the level of a single form to constraint across an entire workflow spanning multiple linked records (Contact, Account, Opportunity).

Haug, Staskiewicz, and Hvam's case study of master-data-management strategy inside an international healthcare company [12] sits closer still to this article's argument. It documents governance strategy, not algorithmic cleanup, as the determinant of master-data outcomes in practice. The guided-intake pattern described here applies that governance argument at the scale of a single transaction rather than as an organization-wide policy.

2.3 Digital Experience Platforms and the provisioning gap

Architecture research on customer-facing digital platforms tends to treat identity provisioning as solved: the portal receives a record, grants access, and the interesting problems live elsewhere. Studies of big-data maturity [13] and CRM digital transformation [14] address data and process maturity broadly, but neither connects specifically to the moment a portal account is provisioned from a CRM record. That is a real gap. A duplicate Contact stops being a

CRM hygiene defect the instant a Digital Experience Platform sits on top of it - it becomes a failed login, a missing application, or a payment attached to the wrong record. Work on API-driven organizational agility [15] and empirical studies of web-service integration practice [16] cover the integration layer in general terms, without examining this specific identity-to-portal handoff. That handoff is the architectural seam this article's design is built to protect.

3. Methodology

This article follows a case-based architectural analysis: an artifact - the guided intake workflow - is described in enough implementation detail to be evaluated and reused elsewhere, its design rationale is made explicit against the problem it targets, and its outcomes are reported as practice-based observations rather than results from a controlled experiment. The case draws on direct architectural and development work on a cloud CRM platform that supports retail and brokered loan origination, with a digital experience platform serving as the downstream consumer of the identity records produced by the workflow.

Two limitations belong here, not buried in a later section. First, this study is a single-organization case. No claim is made that the specific figures below transfer without adaptation to a different CRM implementation or a different industry. Second, the outcomes in Section 6 are the case-based estimates from implementation experience - not measurements against a held-out comparison group.

4. System Architecture and Workflow Design

The guided intake workflow - referred to internally as a “Start Loan” wizard - replaces unconstrained lead conversion with a sequence of screens, each forcing an explicit decision before the next becomes available.

Screen one handles structured file intake. The loan officer uploads a data file, generated by a separate sales-terms tool, containing borrower and loan-term fields. Four borrower-identifying fields - first name, last name, phone, email - are required before any search can run. Proceeding with an under-specified identity simply is not possible.

Screen two enforces search-before-create. Using those required fields, the system searches existing Contact records first, then Lead records if no Contact matches. New-record creation is permitted only once both searches come back empty. This ordering is the single highest-leverage rule in the entire design, since it is exactly the step the legacy process skipped.

Screen three forces the entity decision. Where a matching Contact exists, the loan officer chooses among three explicit paths: link the Contact to its existing Account; link the Contact to a newly created Account, for cases where the same borrower transacts under a different entity; or proceed with a temporary Account when entity information is not yet available. There is no fourth, implicit option.

Screen four routes on ownership. If the matched Contact or Lead belongs to a different owner, the system routes to a request-transfer flow instead of allowing a silent duplicate to appear. This single decision targets what the legacy process treated as an edge case but which, in practice, drove much of the duplicate creation - faced with a record owned by someone else, users would simply create a new one rather than navigate a manual transfer request.

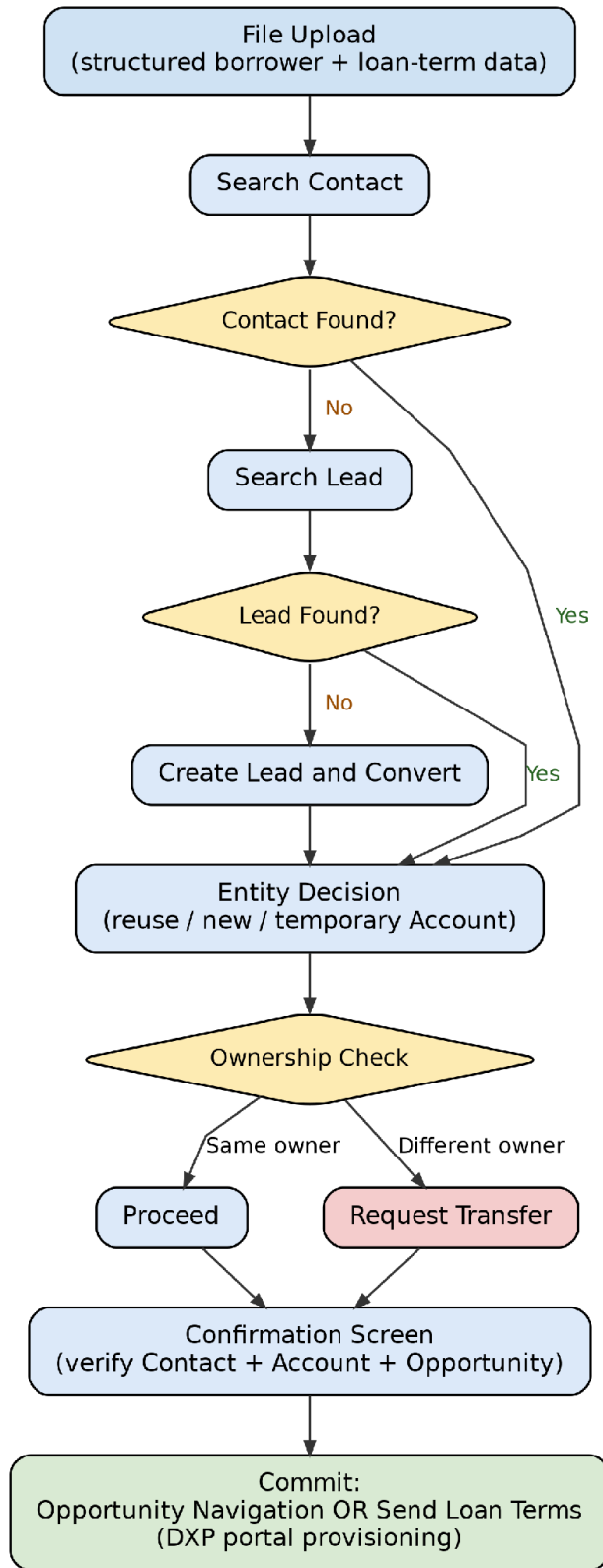
Screen five requires confirmation before commit. Before any record is created or modified, the loan officer reviews a summary of the Contact, Account, and intended Opportunity, and must explicitly confirm before the system proceeds to either Opportunity navigation or the terms-delivery process that provisions Digital Experience Platform portal access.

Table 1. Screen-by-Screen Workflow Logic (Guided Intake Wizard)

Screen	System Behavior	Decision Point
1 - File Intake	Requires four borrower fields (first name, last name, phone, email) before search can run	Cannot proceed with under-specified identity

2 - Search-Before-Create	Searches Contact first, then Lead if no Contact match	New-record creation permitted only if neither exists
3 - Entity Decision	Presents three explicit linkage options	Reuse Account / create new Account / temporary Account
4 - Ownership Routing	Checks record owner against current user	Same owner: proceed. Different owner: request-transfer flow
5 - Confirmation	Displays Contact, Account, and Opportunity summary	Explicit confirm required before commit

Figure 1. Wizard Decision Flowchart



Brokered loans add one screen - broker-entity and broker-contact selection - folded into the same search-before-create discipline, so the pattern extends to a second party type without opening a parallel, less-constrained path.

5. Enterprise Integration Extension

The workflow above still depends on one human-mediated step: a file exported from a separate sales-terms tool and uploaded manually. That is structured intake, but it is not yet integration in the fuller sense of programmable, authenticated, cross-system connectivity [15], [16], [17], [18]. The extension proposed here replaces that manual handoff with an API contract. When loan terms are finalized in the originating tool, that tool posts a structured data package directly to an integration endpoint, authenticated through OAuth client credentials against a dedicated integration service account. The intake wizard opens pre-populated, and the loan officer's attention shifts fully onto the decisions the workflow exists to force - search results, entity choice, ownership routing - rather than onto file mechanics.

Table 2. Legacy File-Based Process vs. Redesigned API-Integrated Process

Dimension	Legacy (File-Based)	Redesigned (API-Integrated)
Data transfer method	Manual CSV export and upload	Authenticated API post (OAuth client credentials)
Manual steps	Export, upload, manual field mapping review	None - wizard opens pre-populated
Validation point	None prior to upload	Schema validation and idempotency at integration boundary
Failure fallback	N/A (manual process)	Manual CSV remains fallback for outages

This matters for the article's central claim in a specific way: prevention-oriented workflow design is only as reliable as the data quality of what enters it. Schema validation and idempotency enforced at the API boundary form a second, earlier control point - upstream of the search-before-create logic itself. That distinction is easy to lose if integration gets treated as a convenience feature rather than as a data-quality control in its own right.

The service-oriented reference architecture formalized by OASIS [19] frames this kind of endpoint as a service contract, not a file drop - a distinction that matters for governance, since a contract can be versioned, validated, and audited in ways a manually uploaded file cannot. Serrano and Zorrilla's layered data-governance framework [20] makes a parallel point at the organizational level: governance quality depends on where controls sit in a data pipeline, not only on whether controls exist somewhere. Read together, these two works support treating the API boundary proposed here as a governance layer in its own right, not merely a faster version of the same file transfer.

6. Results

Reported here as case-based estimates from direct implementation experience (see Section 3, not results from a controlled experiment): average time to create a retail loan record fell from roughly 20 minutes to roughly 2 minutes. Duplicate Account/Contact creation, tracked informally through support-ticket categorization, declined by approximately 90%. Support tickets tied to duplicate records or portal-login failures fell by a comparable margin, around 90%.

These numbers are directional evidence from one implementation, not a validated effect size. One limitation deserves stating plainly: the pre-redesign baseline was reconstructed from informal training-time and ticket-volume records, not captured through a formal instrumented baseline set up in advance.

Table 3. Before/After Case Metrics

Metric	Before Redesign	After Redesign
--------	-----------------	----------------

Average time to create a retail loan record	approximately 20 minutes	approximately 2 minutes
Duplicate Account/Contact creation rate	baseline (unreduced)	reduced by approximately 90%
Support tickets (duplicates / portal login)	baseline (unreduced)	reduced by approximately 90%

7. Discussion

The guided-intake pattern here does not compete with entity-resolution algorithms. It changes how much work those algorithms have left to do. A CRM instance producing fewer duplicates at intake gives blocking and matching pipelines [4], [5] a cleaner substrate to operate on, and leaves a smaller, likely more tractable residual matching problem. Prevention and resolution read as complementary control points under this framing rather than substitutes for one another - a framing this article makes explicit where Haug et al.'s governance-strategy case [12] leaves it implicit.

The Digital Experience Platform dimension is where the stakes stop being abstract. A duplicate Contact is a data-quality defect in the sense the frameworks in Section 2.2 describe [8], [9], [10]. It becomes a customer-facing incident the moment a portal account gets provisioned against the wrong or duplicate record. DXP architecture literature, this article argues, underweights that handoff - treating portal provisioning as a solved downstream step rather than as the point where upstream identity failures surface as something a customer actually experiences.

Saha and colleagues' systematic review of CRM and data-analytics integration [21], [22], [23] offers a useful caution here: analytics built on top of a CRM are only as trustworthy as the identity records feeding them, and a review focused on analytics maturity can understate how much of that trustworthiness is decided upstream, at intake, before any analytics pipeline ever touches the data. This article's argument extends that caution into a specific design recommendation rather than leaving it as an implicit precondition [24], [25].

Two limitations bound these claims. The case comes from one organization and one vertical - mortgage lending - so the specific screen sequence would need adaptation elsewhere, though the underlying principles (search before create, explicit entity decisions, ownership-aware routing) are not lending-specific and nothing in the design depends on loan-industry particulars. And as Section 6 notes, the outcome figures are practice-based estimates rather than results from an instrumented controlled study [24], [26].

8. Conclusion

Duplicate identity records in enterprise CRM are not purely a downstream matching problem. A meaningful share of them start as a workflow-design problem, created at the moment of intake by processes that allow record creation without first requiring a search. Guided, search-before-create workflow design - paired with explicit entity-linking decisions, ownership-aware routing, and an integration layer that removes manual file handling - is an underused complement to entity-resolution algorithms. That complementarity is most visible at the Digital Experience Platform layer, where identity defects stop being an internal hygiene concern and become something a customer runs into directly. The pattern here travels beyond its originating case. Any cloud CRM architecture where a defined business process creates customer records is a candidate for the same prevention-first redesign.

9. Declarations

Funding: This research received no external funding.

Author Contribution: R. C. Chinta is the sole author and is responsible for all aspects of this work.

Conflict of Interest: The author declares no conflict of interest.

Data Availability: The case-based outcome estimates reported in this article derive from the author's professional implementation experience and do not constitute a shared dataset.

REFERENCES

1. I. P. Fellegi and A. B. Sunter, "A Theory for Record Linkage," *Journal of the American Statistical Association*, vol. 64, no. 328, pp. 1183-1210, 1969. <https://www.cs.cornell.edu/~shmat/courses/cs6434/fellegi-sunter.pdf>
2. V. Christophides, V. Efthymiou, T. Palpanas, G. Papadakis, and K. Stefanidis, "An Overview of End-to-End Entity Resolution for Big Data," *ACM Computing Surveys*, vol. 53, no. 6, art. 127, 2021. <https://dl.acm.org/doi/abs/10.1145/3418896>
3. B.-H. Li, Y. Liu, A.-M. Zhang, W.-H. Wang, and S. Wan, "A Survey on Blocking Technology of Entity Resolution," *Journal of Computer Science and Technology*, vol. 35, no. 4, pp. 769-793, 2020. <https://link.springer.com/article/10.1007/s11390-020-0350-4>
4. S. Thirumuruganathan, H. Li, N. Tang, M. Ouzzani, Y. Govind, D. Paulsen, G. Fung, and A. Doan, "Deep Learning for Blocking in Entity Matching: A Design Space Exploration," *Proceedings of the VLDB Endowment*, vol. 14, no. 11, pp. 2459-2472, 2021. <https://dl.acm.org/doi/10.14778/3476249.3476294>
5. F. Neuhoof, M. Fisichella, G. Papadakis, K. Nikolettos, N. Augsten, W. Nejdl, and M. Koubarakis, "Open benchmark for filtering techniques in entity resolution," *The VLDB Journal*, vol. 33, no. 5, pp. 1671-1696, 2024. <https://dl.acm.org/doi/abs/10.1007/s00778-024-00868-7>
6. Ravikanth M, S. Korra, G. Mamidiseti, M. Goutham, and T. Bhaskar, "An efficient learning based approach for automatic record deduplication with benchmark datasets," *Scientific Reports*, vol. 14, art. 16254, 2024. <https://www.nature.com/articles/s41598-024-63242-1>
7. O. Binette and R. C. Steorts, "(Almost) all of entity resolution," *Science Advances*, vol. 8, no. 12, eabi8021, 2022. <https://www.science.org/doi/10.1126/sciadv.abi8021>
8. R. Y. Wang and D. M. Strong, "Beyond Accuracy: What Data Quality Means to Data Consumers," *Journal of Management Information Systems*, vol. 12, no. 4, pp. 5-33, 1996. <https://www.tandfonline.com/doi/abs/10.1080/07421222.1996.11518099>
9. J. Wang, Y. Liu, P. Li, Z. Lin, S. Sindakis, and S. Aggarwal, "Overview of Data Quality: Examining the Dimensions, Antecedents, and Impacts of Data Quality," *Journal of the Knowledge Economy*, vol. 15, no. 1, pp. 1159-1178, 2024. https://www.researchgate.net/publication/368423995_Overview_of_Data_Quality_Examining_the_Dimensions_Antecedents_and_Impacts_of_Data_Quality
10. R. Miller, H. Whelan, M. Chrubasik, D. Whittaker, P. Duncan, and J. Gregorio, "A Framework for Current and New Data Quality Dimensions: An Overview," *Data*, vol. 9, no. 12, art. 151, 2024. https://www.researchgate.net/publication/387205180_A_Framework_for_Current_and_New_Data_Quality_Dimensions_An_Overview
11. M. Breitmayer, L. Arnold, M. Winter, and M. Reichert, "Exploring user editing preferences in electronic forms: an empirical study," *Information Systems and e-Business Management*, vol. 22, no. 1, pp. 171-207, 2024. <https://link.springer.com/article/10.1007/s10257-023-00662-y>
12. A. Haug, A. M. Staskiewicz, and L. Hvam, "Strategies for Master Data Management: A Case Study of an International Hearing Healthcare Company," *Information Systems Frontiers*, vol. 25, no. 5, pp. 1903-1923, 2023. <https://link.springer.com/article/10.1007/s10796-022-10323-z>
13. Z. A. Al-Sai, M. H. Husin, S. M. Syed-Mohamad, R. Abdullah, R. Abu Zitar, L. Abualigah, and A. H. Gandomi, "Big Data Maturity Assessment Models: A Systematic Literature Review," *Big Data and Cognitive Computing*, vol. 7, no. 1, art. 2, 2023. <https://www.mdpi.com/2504-2289/7/1/2>
14. H. Gil-Gomez, V. Guerola-Navarro, R. Oltra-Badenes, and J. A. Lozano-Quilis, "Customer relationship management: digital transformation and sustainable business model innovation," *Economic Research-Ekonomska Istrazivanja*, vol. 33, no. 1, pp. 2733-2750, 2020. https://www.researchgate.net/publication/338725465_Customer_relationship_management_digital_transformation_and_sustainable_business_model_innovation
15. J. Ofoeda, R. Boateng, and J. Effah, "API integration and organisational agility outcomes in digital music platforms: A qualitative case study," *Heliyon*, vol. 10, no. 11, e31756, 2024. <https://www.sciencedirect.com/science/article/pii/S2405844024077879>
16. S. Baravkar, P. Gaikwad, Z. Li, E. Tilevich, L. Cheng, and Z. Song, "Towards a comprehensive understanding of web service integration: a large-scale empirical study from the developers' perspective," *Empirical Software Engineering*, vol. 30, no. 5, art. 144, 2025. <https://link.springer.com/article/10.1007/s10664-025-10647-9>
17. R. Sun, Q. Wang, and L. Guo, "Research Towards Key Issues of API Security," in *Cyber Security: 18th China Annual Conference, CNCERT 2021, Revised Selected Papers, CCIS vol. 1506*, Springer, 2022, pp. 179-192. https://www.researchgate.net/publication/358011553_Research_Towards_Key_Issues_of_API_Security
18. I. Odun-Ayo, M. Adebisi, O. Okesola, and O. Vincent, "Cloud middleware and services - a systematic mapping review," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 5, pp. 2696-2706, 2021. <https://beei.org/index.php/EEI/article/view/3169>

19. OASIS, "Reference Model for Service Oriented Architecture 1.0," OASIS Standard, 2006. https://www.researchgate.net/publication/228403638_Reference_model_for_service_oriented_architecture_10
20. J. Y. Serrano and M. Zorrilla, "A Data Governance Framework for Industry 4.0," IEEE Latin America Transactions, vol. 19, no. 12, pp. 2130-2138, 2021. <https://ieeexplore.ieee.org/document/9480156>
21. L. Saha, H. K. Tripathy, S. R. Nayak, A. K. Bhoi, and P. Barsocchi, "Amalgamation of Customer Relationship Management and Data Analytics in Different Business Sectors - A Systematic Literature Review," Sustainability, vol. 13, no. 9, art. 5279, 2021. <https://www.mdpi.com/2071-1050/13/9/5279>
22. H. Gaggar, "MemGov: Policy-Governed Memory Lifecycle Management for Enterprise LLM Agents," Journal of Intelligent Decision Making and Information Science, vol. 3, no. 11s, pp. 2199-2214, 2026.
23. K. Rekulapalli, H. Nilagiri, and M. Vakkanthula, "A Comparative Evaluation of National FHIR Implementation Guides: Strategic Methodologies for Enhancing Interoperability, Regulatory Alignment, and Clinical Comprehensiveness in Global Health IT," Health Systems, pp. 1-13, 2026. <https://doi.org/10.1080/20476965.2026.2728621>
24. [24]A. Bhatnagar and V. R. K. Bonagiri, "Operational Excellence through Cloud-Native Distributed Systems," *International Journal of Computational and Experimental Science and Engineering*, vol. 12, no. 2, 2026. <https://doi.org/10.22399/ijcesen.5207>
25. H. Gaggar, "Architecting Scalable Enterprise AI Agent Infrastructure for Distributed Intelligent Systems," International Journal of Advances in Signal and Image Sciences, pp. 193-208, 2025. <https://doi.org/10.29284/da9w0117>
26. C. T. Bobba, A. G. Vijay, V. Aher, J. Singh, A. Kejriwal, K. Srinivasan, and A. Ranjan, "Radiographic Evaluation of Periapical Healing Following Single vs Multiple Visit Root Canal Treatment: A PAI Based Study," Journal of Pharmacy & Bioallied Sciences, vol. 18, Suppl. 1, art. S206, 2026.