From Catalog Fragmentation to Procurement Visibility: An Agentic Reference Architecture for Puerto Rico's First Government Item Master

Jared Villanueva Valentín
Master of Engineering in Computer Engineering
Advisor: Nelliud D. Torres Batista, DBA
Polytechnic University of Puerto Rico
Graduate Project EXPO, May 2026

Abstract — *The General Services Administration (ASG) is Puerto Rico's central procurement agency, yet its catalog contains hundreds of thousands of records with extensive duplication: a single blue ballpoint pen appears in over a hundred textual variations. This paper presents the design and implementation of an agentic Big Data infrastructure that converts the fragmented catalog into a controlled single source of truth. The architecture combines a Databricks Medallion Lakehouse with Microsoft Foundry agents that produce standardized product names and assign National Institute of Governmental Purchasing (NIGP) codes. Delivered artifacts include a daily ingestion task, cleansing and deduplication logic, five Foundry agents with responsible-AI safeguards, a unique item identifier, and a Minimum Viable Data template. The as-deployed first-year cost (~$72,000 USD) is roughly an order of magnitude below earlier hand-curation quotes. The result is a governance-aligned reference architecture for public-sector catalog modernization.*

Keywords — *Agentic AI, Big Data, Item Master, Public Procurement.*

INTRODUCTION

In July 2025, the General Services Administration (ASG) of Puerto Rico issued a Request for Information for AI solutions integrated across agencies and municipalities [1]. The RFI required solutions that were viable, secure, and ethical, signaling a clear governmental focus: modernizing public administration through data-driven, AI-assisted decision-making.

The operational and legal framework is already in place. The Puerto Rico procurement system was reformed through *Ley 73-2019*, which made ASG the central authority in the acquisition of all goods and services used by the Government of Puerto Rico [2], and its implementing regulation, *Reglamento Núm. 9292*, codifies the procedural framework with which agencies work [3]. JEDI, the Government of Puerto Rico's procurement platform through which all purchases must pass, is the day-to-day operational system, and Oracle Cloud Fusion ERP is the planned financial system of record.

But the gap is structural. The Government of Puerto Rico has no authoritative record of what it purchases. The catalog distributed via JEDI is fragmented to the point that it cannot be audited, cannot support useful spend analytics, and cannot support the AI-driven decision-making that the larger strategy relies on. Even the agencies that ASG is mandated to modernize under *Ley 73-2019* [2] cannot answer basic questions about what they have already purchased.

This article describes the design and implementation of an agentic Big Data infrastructure for the ASG procurement catalog: a Databricks Medallion Lakehouse that ingests data from a copy of the JEDI procurement database, cleanses and deduplicates it, and publishes a governed Item Master as timestamped CSV and NDJSON extracts on Azure Data Lake Storage Gen2 for downstream consumption by external tools. I designed the end-to-end Databricks-on-Azure architecture and implemented the Bronze ingestion tier, the Silver-tier cleansing and two-pass deduplication logic, the Gold-tier curation and ASG-ITEM-ID assignment, and the daily orchestration. The initial diagnostic analysis of the JEDI extracts and the Microsoft Foundry agent layer were contributed by Social Glass, Inc. engineering personnel [4], [5], in alignment with the architecture and operational behavior described in this paper. The contribution is a reproducible producer-only architecture pattern for public-sector item-master modernization that combines Lakehouse data engineering with LLM-agent enrichment under measurable responsible-AI constraints. In this work, "agentic" refers to a multi-agent orchestration pattern in which specialized LLM-driven components execute bounded classification and enrichment tasks under explicit governance constraints.

BACKGROUND

Big Data and AI in Public Procurement

The 2025 OECD analysis identifies procurement as a high-leverage AI application (well-organized supplier and item data, repetitive transactional decisions, clear audit demand) but cautions that public-sector AI must be transparent, auditable, and bounded by data governance [6]; the responsible-AI literature also warns that generative models produce confident outputs irrespective of underlying support [7]. The system here must therefore measurably improve catalog quality while remaining auditable, reversible, and governed.

Entity Resolution and Catalog Quality

The technical core of the work is entity resolution: many records refer to the same product under different textual representations. The duplicate-record-detection survey by Elmagarmid et al. defines the standard pipeline of blocking (coarse-key candidate restriction), similarity scoring (character- or token-based), and survivorship (golden-record selection) [8]. The literature also documents the false-positive / false-negative trade-off that governs threshold selection. The approach used here applies MinHash LSH on 3-grams of the description with brand-and-model blocking, followed by Levenshtein verification, with the longest cleaned description chosen as survivor.

PROBLEM

The diagnostic stage run on a snapshot of the JEDI procurement system, demonstrated a catalog with serious scale and quality issues [5]. The raw extract contained 299,986 records, of which 39,616 had identical descriptions: the same product listed with the same description but different internal identifiers, sometimes by different agencies on different dates. Beneath that exact-match layer there was a much larger range of near-duplicates exhibiting lexical variation. One blue ballpoint pen, one of the most common office supplies in any government, appeared in well over a hundred different textual representations, with variations in capitalization, accentuation, brand-name position, and abbreviation. This 117-variant pen is the most basic diagnostic of the larger issue: when a common office item cannot be counted reliably, the catalog itself cannot reliably inform spend analytics, audit, or AI-driven decisions for any other item.

The fragmentation goes beyond text variation. A field-level audit of the same extract reveals that more than 70% of records lack key descriptive fields — brand, model, unit of measure, contract number, warranty — that would otherwise allow distinct items to be told apart by structured comparison alone. It is this missing-fields problem that drives the description-level entity-resolution approach used further down the pipeline.

These organizational side effects of fragmentation cut across all functions the catalog was intended to perform. Spend analytics fail because the same purchase is not counted as one purchase; auditability is undermined because the unit of analysis is volatile; interoperability with state and federal taxonomies — NIGP for procurement classification and NAICS for industry classification — fails because the underlying records cannot be consistently mapped. The current catalog allocates 12 top categories across 75 subcategories, but those categories are themselves polluted by the same fragmentation and cannot be relied upon as analytic boundaries. Adoption of the National Institute of Governmental Purchasing (NIGP) taxonomy resolves this fragmentation by replacing the legacy 12-category / 75-subcategory partition with a richer, federally-aligned classification of approximately 1,957 sub-categories, restoring interoperability with

U.S. state and federal procurement systems. The consequence is that ASG cannot answer the most fundamental procurement question — what the Government of Puerto Rico actually purchases — and therefore cannot operationalize the AI-driven decision-making that the government-wide AI strategy depends on.

The article has four implementation objectives:

1. Create a unique, traceable item identifier compatible with the JEDI procurement system and Oracle Cloud Fusion ERP, with simplified variants for requisitions and solicitations.
2. Build a cleansing-and-deduplication pipeline on a Databricks Medallion lakehouse that scales to the entire ASG catalog.
3. Implement NIGP classification with AI assistance under explicit responsible-AI safeguards.
4. Define a governance model that maintains the catalog as a single source of truth across the participating agencies over time.

METHODOLOGY

The methodology traces the path of a record through the system from raw extraction in the JEDI copy database to publication of the cleansed, enriched Item Master as CSV and NDJSON extracts on ADLS Gen2. Figure A1 shows the architecture, and the subsections that follow detail one step of the pipeline at a time.

Pipeline Architecture

The architecture is a three-stage Medallion lakehouse on Azure Databricks [9], [10] augmented with Microsoft Foundry agents [11] over Azure OpenAI Service. Stages exchange timestamped CSV/NDJSON extracts on ADLS Gen2 ; each Databricks cluster terminates after its Job. Two cross-cutting services apply to every stage: Databricks Secrets (credentials, with no plaintext in any notebook) and Unity Catalog [10] (lineage and access control). Figure 1 shows the components.

Commented [NT1]: Entiendo que esta parte debe seguir hacia abajo, no hacia al lado.

Commented [JV2R1]: Addressed & Implemented

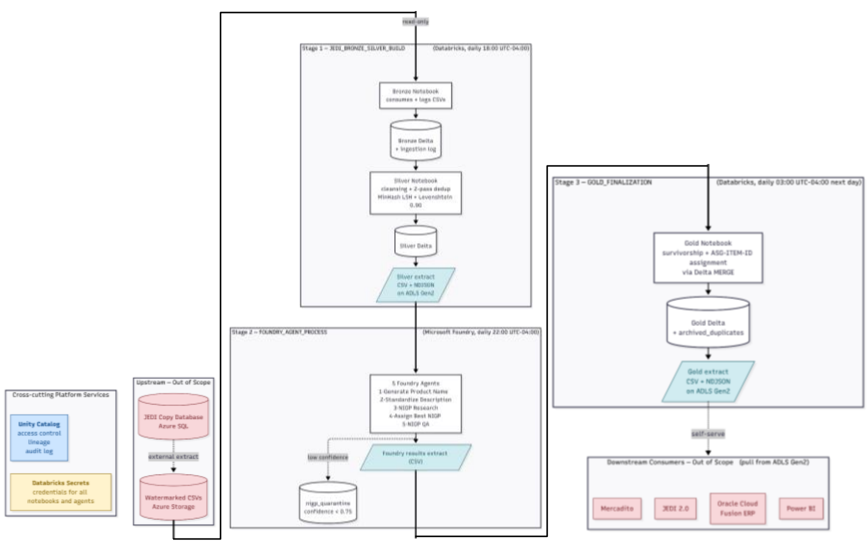

Figure 1

End-to-End Architecture: Medallion Lakehouse on Azure Databricks with Microsoft Foundry Agents

Commented [NT3]: Entiendo que la imagen es demasiado larga, pero no se si se permita cruzarla en dos columnas. Recomiendo partirla en la parte A y B y podrías ponerla en una sola columna.

Commented [NT4R3]: Tampoco sepuede leer bien su contenido. Si cada diagrama pudiese tener una tiutlo en grande podría ayudar. Otra sugerencia es ponerla como un apéndice al final del artículo y usar una página completa para que se pueda mostrar bien.

Commented [JV5R3]: Addressed & Implemented

Stage 1 — Bronze Ingestion and Silver Cleansing

Stage 1 runs as a single Databricks Job containing two PySpark/Python notebooks (Bronze and Silver), executing daily at 18:00 (UTC-04:00) on a General Cluster with Databricks Runtime 16.4 LTS (Apache Spark 3.5.2, Scala 2.12), Standard_D4ds_v5 driver and worker nodes auto-scaling between two and eight workers.

The Bronze notebook records raw procurement information with little processing, maintains the original audit trail, and decouples downstream processing from the upstream system. It consumes watermarked, date-stamped CSV files from an Azure Storage container and loads them into the Bronze Delta table. Both the JEDI procurement database copy (Azure SQL) and the upstream extraction job that writes the CSVs to storage are maintained outside the scope of this work; this read-only boundary insulates the live procurement system from any pipeline activity. Append semantics ensure historical data is never overwritten. The ingestion is safely repeatable: a companion log table records every CSV processed, and each run only ingests files not already in the log. To remain resilient to flawed source data, the CSV reader runs in permissive mode with multi-line and escaped-quote options, allowing embedded line breaks and quoted commas inside item descriptions to pass without rejecting the row.

The Silver notebook then converts raw Bronze records into a cleansed, semantically typed, reference-mapped table that is ready for deduplication. The build first selects the columns needed downstream (item, description, brand, model, agency, acquisition method, date) and runs an initial exact match deduplication. The cleanup pass then does most of the work: tokens such as “n/a”, “n\a”, ”na”, “no aplica”, “x”, isolated dots(“.”), and dash-only(“-“) entries are replaced with empty strings via case-insensitive regex, whitespace is trimmed, and a second exact-match deduplication pass collapses records that differed only by a placeholder. Type casting then converts date columns from strings to timestamp/datetime, casts numeric columns to their appropriate types, and removes trailing zeros from source formatting (for example, 5.00 becomes 5). Finally, two small lookup tables (Agency-ID and Acquisition-Method) convert human-readable agency names and procurement-method strings into stable short codes that feed the corresponding segments of the ASG-ITEM-ID.

Deduplication in the Silver tier proceeds in two passes [8]:

- **Pass 1 — Exact-match deduplication.** dropDuplicates() are applied twice inside the cleanup logic above (once before placeholder normalization and once after), to collapse records that differ only by tokens like “n/a” or dash-only entries(“-“).
- **Pass 2 — Near-duplicate resolution (two-phase hybrid).**
  - Phase A — Candidate generation. MinHash Locality-Sensitive Hashing (LSH) on 3-gram tokens of the normalized description, with exact blocking on brand and model when both are populated; recall-favoring, surfaces candidate pairs cheaply.
  - Phase B — Verification. Levenshtein edit distance is computed for each candidate pair, converted to a similarity score, and pairs with similarity greater than 0.90 are confirmed as duplicates.

The two thresholds are tuned independently. The LSH phase controls how aggressively candidates are surfaced; the Levenshtein phase controls how strictly they must match before merging.

The 0.90 cut-off was tuned against a hand-labeled validation set drawn from the diagnostic corpus. The 117 ballpoint-pen variants all scored above the threshold against one another, while distinct products scored visibly below; on the labeled subset, the chosen threshold favored under-merging over the more damaging failure mode of false merges [8], which silently corrupt downstream analytics and are not easily reversed. Full-corpus

Commented [NT6]: El guión no va...

Commented [JV7R6]: Addressed & Implemented

Commented [NT8]: Entiendo debe ir entre paréntesis

Commented [JV9R8]: Addressed & Implemented, separated each token clearly using quotation marks

Commented [NT10]: Cuál es la palabra que quieres poner aquí??

Commented [JV11R10]: Addressed & Implemented, completed the word “deduplication” precision/recall characterization is left for future work.

Stage 2 — Microsoft Foundry Agents

Built on Microsoft Foundry's Agent Service [11], the agentic enrichment layer executes large language models as goal-directed agents over Azure OpenAI Service: gpt-5-nano powers each agent except Assign Best NIGP, which uses gpt-5-mini for the higher-stakes classification step. Four production agents (Generate Product Name, Standardize Description, NIGP Research, Assign Best NIGP) operate between the Silver and Gold tiers, with a fifth agent, NIGP QA, validating each NIGP-5 assignment downstream of Assign Best NIGP. The agents function under responsible AI constraints: every output carries a confidence score and is both auditable and reversible. Records with NIGP confidence below 0.75 are routed to a nigp_quarantine table for downstream review; the steward review, correction, and reinjection workflow that closes the loop on quarantined records is left for future work.

The Generate Product Name and Standardize Description agents operate only on input attributes that exist; they never produce or impute missing specifications. This avoids the known hallucination risks of large language models in structured tasks. The NIGP Research and Assign Best NIGP agents work strictly within the official NIGP taxonomy (licensed from Periscope Holdings or an equivalent NIGP licensor) and do not synthesize codes, so the assignments interoperate with state and federal procurement systems that use the same standard.

The 0.75 confidence cut-off is the main quality gate: low-confidence outputs are routed to quarantine instead of being committed. The threshold was selected to bias against false-positive classification on the validation subset; tighter calibration against full-corpus precision/recall, together with the steward review-and-reinjection loop, is left for future work.

Stage 3 — Gold Curation and Publication

The Gold tier, executed by a PySpark/Python Databricks notebook in Stage 3, consumes the Foundry-enriched records leaving the agent layer and produces the canonical Item Master. Within each cluster of duplicates resolved upstream in the Silver tier, the record with the longest cleaned description is selected as the golden record, on the assumption that longer descriptions carry more specifying detail. Non-surviving records remain linked to the golden record through an archived_duplicates table for historical traceability. The pipeline degrades gracefully on records with missing metadata (for example, supplier submissions that do not include complete brand or model); incomplete records are still processed rather than failing the run.

Each golden record is assigned a unique ASG-ITEM-ID per the encoding rules in the next subsection. The resulting item_master_gold Delta table is the canonical Item Master that downstream consumers read; each publication is tagged with a pipeline-run identifier and timestamp recoverable through Delta time travel, and the table is never destructively overwritten.

ASG-ITEM-ID Encoding

Objective 1 requires a unique, traceable identifier for an individual item that can be used in JEDI and in Oracle Cloud Fusion ERP. The selected format is a hyphenated five-segment string of the form AAA-BB-CCC-DD-EEEEEEEEEE. Table 1 enumerates each segment, its width, what it encodes, and the source of its value within the pipeline. Each identifier is therefore self-explanatory: when read, it conveys the procurement category, agency, acquisition method, and relative order within the catalog without requiring an external lookup.

Table 1
ASG-ITEM-ID Segments

Segment	Width	Position	Encodes	Source
AAA	3 digits	1	NIGP class (top-level procurement category)	NIGP QA Agent (Mode 1)
BB	2 digits	2	NIGP item (sub-category within the class)	NIGP QA Agent (Mode 1)
CCC	3 digits	3	ASG-internal agency code	Silver-tier reference join (agency_dim)
DD	2 digits	4	Acquisition method (01–06; see list below)	Silver-tier reference join (acquisition_method_dim)
EEEEEEEEEE	10 digits	5	Globally unique chronological sequence	Gold-tier transactional Delta MERGE

The acquisition-method segment encodes the six procurement modes recognized under Puerto Rico's regulatory framework:

- 01 contract purchase (compra contra contrato)
- 02 direct purchase (compra directa)
- 03 public bidding (licitación pública)
- 04 emergency purchase (compra de emergencia)
- 05 cooperative purchase (compra cooperativa)
- 06 exception purchase (compra por excepción).

By incorporating the method into the identifier itself, spending analytics across acquisition mode become trivially queryable and do not need a join with transactional metadata, which may have different access policies.

Two simplified variants accommodate procurement workflows where the complete identifier is not yet available. The requisition variant takes the form AAA-BB-EEEEEEEEEE, dropping the agency and acquisition-method segments because those values are not determined until the requisition is converted into a purchase order. The solicitation variant is of the form AAA and encodes only the NIGP class, because at the bid-solicitation point the item of interest is deliberately underspecified to allow multiple suppliers to respond. Both variants resolve to the complete ASG-ITEM-ID at purchase-order creation in JEDI, when all five segments become known.

System Integration

The pipeline publishes the managed Item Master as timestamped CSV and NDJSON extracts on Azure Data Lake Storage Gen2 (extracts/gold/gold_extract_*.{csv,ndjson}); each run accumulates rather than overwriting, so consumers can replay any historical state. The Bronze, Silver, and Gold Delta tables are also queryable through the Databricks SQL endpoint on a manually-started cluster. Consumption is pull-based and out of scope: external systems (JEDI 2.0, Oracle Cloud Fusion ERP, Mercadito [12], Power BI) connect to ADLS Gen2 and ingest the standardized columns (ASG-ITEM-ID, item name, description, NIGP class/item, UOM, acquisition method) on their own schedule. The pipeline does not push to or coordinate with any downstream consumer.

Production Scheduling and Operations

Table 2 summarizes the three scheduled stages, each connected by extract artifacts on ADLS Gen2. Stages are independently orchestrated rather than coupled: Stage 2 runs ~4 hours after Stage 1, and Stage 3 runs ~5 hours after Stage 2, so the Gold extract is available before the 06:00 UTC-04:00 daily SLA. Each stage is idempotent: transient failures auto-retry, persistent failures escalate for human attention.

Commented [NT12]: No veo el titulo de la tabla y el texto está muy pegado. Entiendo que se puede tratar de poner en una columna o como un segundo apéndice...

Commented [JV13R12]: Addressed & Implement

Commented [NT14]: Verificar si va spending en lugar de spend

Commented [JV15R14]: Addressed and implemented

Table 2
Three Pipeline Stages

Stage	Job / Process	Schedule	Inputs	Outputs	Cluster
1	JEDI_BRONZE_SILVER_BUILD (Databricks)	Daily 18:00 UTC-04:00	JEDI copy-DB CSV files	Bronze Delta + ingestion log; item_master_silver Delta; Silver extract (CSV + NDJSON)	Standard_D4ds_v5, 2–8 workers; terminates after Job
2	FOUNDRY_AGENT_PROCESS (Microsoft Foundry)	Daily 22:00 UTC-04:00	Latest Silver extract	Foundry results extract (CSV); nigp_quarantine for low-confidence records	External; no Databricks cluster
3	GOLD_FINALIZATION (Databricks)	Daily 03:00 UTC-04:00 (next day)	Latest Foundry results extract; reference dimensions	item_master_gold Delta + archived duplicates; Gold extract (CSV + NDJSON)	Standard_D4ds_v5, 2–8 workers; terminates after Job

Commented [NT16]: Lo mismo con esta tabla....el orden del articulo es dificil de seguir...

Commented [JV17R16]: Addressed & Implemented

Governance

Three governance dimensions support the catalog as a single source of truth: Platform, Process, and Operational.

Platform Governance

Unity Catalog [10] is the platform-layer system of record. It enforces table-level and column-level access control (with attribute-based policy on sensitive supplier and financial data), traces end-to-end lineage from Bronze ingestion through Silver curation and Foundry-agent enrichment to Gold publication, and maintains an immutable audit log of every read or write performed against the lakehouse. Every action taken on the catalog is therefore attributable, so the platform itself is the system of record.

Process Governance

Catalog Stewards are designated procurement officers who hold explicit authority over the Item Master, so every record is approved before integration. Supplier intake is anchored to the Minimum Viable Data (MVD) template, which defines the contractual data requirements (eleven required fields plus ten optional ones) every supplier must meet for a submission to be valid.

Operational Governance

Operational measurement spans two KPI tiers. Pipeline KPIs (NIGP QA precision, Generate Product Name compliance, pipeline latency, and deduplication rate) are computable from the produced extracts and are reported in the Discussion. Procurement KPIs are designed for the operational state and require downstream consumer integration to measure; they are specified here as targets the architecture supports: requisition speed (submission-to-approval elapsed time), error rate (percentage of requisitions returned for correction), and duplicate prevention rate (percentage of incoming requisitions that match an existing Item Master record rather than triggering new-item creation).

RESULTS AND DISCUSSION

Project outputs correspond one-to-one with the four implementation objectives outlined in the Problem section.

Item Identifier

The five-part ASG-ITEM-ID specification AAA-BB-CCC-DD-EEEEEEEEEE (Table 1) was defined, validated for compliance with the JEDI and Oracle Cloud Fusion ERP integration requirements, and accepted by the ASG team. The procurement workflow includes two simplified variants (AAA-BB-EEEEEEEEEE for requisitioning and AAA for solicitation) that expand to the complete identifier upon purchase-order generation, per the ASG-ITEM-ID Encoding subsection.

Pipeline

The Bronze ingestion runs daily per the workflow in the Production Scheduling subsection (Table 2). Silver and Gold pipeline components have been developed and run against the full diagnostic snapshot described in the Problem section. All catalog items were processed through cleansing, deduplication, and Foundry-agent enrichment; the evaluation metrics reported below were drawn from two samples: pipeline latency and deduplication rate from the stratified 10% sample, and NIGP QA precision and Generate Product Name compliance from a 50-item manual subset; full-scale measurement of the human-review KPIs is left for future work. The diagnostic snapshot is governed by ASG and is not publicly distributable.

AI Classification

The five Microsoft Foundry agents have been deployed against OpenAI's gpt-5-nano on Azure OpenAI Service and bounded by responsible-AI guidelines: no fabrication of unspecified attributes, confidence-scored outputs, and routing of any output below the 0.75 confidence threshold to a quarantine table; review and reinjection of quarantined records is left for future work.

Governance Framework

The three-dimensional framework described in the Governance subsection is in place; pipeline KPIs are measured and procurement KPIs are defined as targets. A license contract with Periscope Holdings (or an equivalent NIGP licensor) covers the classification taxonomy.

Workplan Execution

Table 3 reports planned versus actual hours by month over the January–June 2026 implementation cycle, with operational handover to ASG scheduled at the end of June.

Table 3
Implementation Workplan and Execution Status

Month	Phase	Est. Hr.	Actual	Status
Jan 2026	Foundation, cleanup, QA design	80	66	Complete
Feb 2026	Classification and code generation	77	70	Complete
Mar 2026	System integration	62	79	Complete (overrun)
Apr 2026	Testing and user acceptance	62	—	Complete
May 2026	Vendor and user alignment	50	—	In progress
Jun 2026	Launch and operational handover	40	—	Pending

Commented [NT18]: Verdaderamente recomiendo utilizar un apéndice

Commented [JV19R18]: Addressed & Implemented

Final empirical results from the testing phase are reported in the Discussion; the remaining May–June milestones focus on operational rollout rather than additional measurement.

Cost Comparison

The agentic architecture also costs substantially less than the earlier proposals to construct the Item Master manually. Two prior estimates priced the work without AI assistance, both based on hand-curation by procurement specialists: NIGP quoted approximately $1,000,000 USD, and Social Glass initially quoted $200,000 USD. The design described in this paper allowed ASG to renegotiate the Social Glass engagement to $63,250 USD for the full scope of both Glass deliverables (the diagnostic analysis [5] and the Foundry agent implementation [4]) plus operational support [13]. Table 4 summarizes the as-deployed cost profile.

Infrastructure costs are minimal because the architecture is pay-as-you-go, clusters terminate after each Job and incur no idle charges (operational costs derived from Azure consumption metering). The worst-case $750–925/month figure applies only to disaster-recovery reprocessing, not to normal daily operation. Under the as-deployed operating model, the first-year all-in cost is approximately $72,000 USD, against the original $200,000–$1,000,000 USD hand-curation quotes.

Table 4
Cost Comparison

Cost Item	Manual Approach (Quoted)	As-Deployed Approach
Item Master construction (NIGP quote, manual)	$1,000,000 (one-time)	—
Item Master construction (Social Glass quote, manual)	$200,000 (one-time)	—
Social Glass engagement (renegotiated; diagnostic [5] + Foundry implementation [4] + ops) [13]	—	$63,250 (one-time)
NIGP taxonomy license (Periscope, via Social Glass)	—	$7,000 / year
Infrastructure setup (Databricks, Azure, storage, Foundry; pay-as-you-go)	—	~$100 (one-time)
Normal monthly operational cost (incremental processing)	—	~$150 / month
Worst-case monthly cost (full historical reprocessing; disaster recovery only)	—	$750–925 / month

DISCUSSION

Four explicit compromises shape the design:

1. **Conservative deduplication threshold (0.90).** The two-level approach (MinHash LSH for candidate generation, followed by Levenshtein verification) accepts a higher probability of missing duplicates in exchange for very few false positives, which would otherwise contaminate the entire downstream process.
2. **Automation vs. validation in NIGP classification.** The agents classify thousands of records without delay but cannot self-correct under uncertainty. The 0.75 confidence cut-off, chosen to bias against false-positive classification, partitions each enriched record into either the accepted set published to the Item Master or nigp_quarantine; the Catalog Steward review-and-reinjection workflow that closes the loop is left for future work.
3. **Governance overhead vs. catalog drift.** The Catalog Steward role and the Minimum Viable Data template create a barrier to indiscriminate data inclusion. This barrier matters. The 39,616 duplicate entries with identical names are the result of poor governance; this overhead is the cost of not letting that happen again.
4. **Responsible-AI constraints adopted from the literature [6], [7]:** output confidence scores, the 0.75 quarantine threshold, and the agents' inability to manufacture any attribute beyond what is explicitly provided.

Empirical evaluation was performed on two samples. Pipeline latency and deduplication rate were computed on a stratified 10% sample (approximately 30,000 records) of the processed corpus. NIGP QA precision and Generate Product Name compliance were measured on a 50-item manual subset, because sampling-based human review at the 10% scale is not tractable under the current scope; full-scale measurement of the human-review KPIs is left for future work. Table 5 reports the pipeline KPIs against the architecture's stated targets. End-to-end wall-clock latency is approximately 9 h, dominated by the scheduled inter-stage gaps in Table 2; active compute across all three stages totals 68–80 min.

Table 5

Pipeline KPIs: Targets and Measured Results (10% Stratified Sample + 50-Item Manual Subset)

KPI	Definition	Target	Measured
NIGP QA precision	% of accepted (≥0.75) NIGP assignments confirmed by human review	≥ 90%	92% (46/50, manual subset)
Generate Product Name compliance	% of outputs in canonical [Generic Name] - [Brand/Model] - [Key Specs] - [UOM] form with no fabricated specifications	≥ 95%	96% (48/50, manual subset)
Pipeline latency	Active processing time from copy-DB extract to Gold extract availability (excludes scheduled inter-stage gaps)	≤ 90 min active compute	68–80 min (full 10% sample run)
Deduplication rate	Reduction in unique product records after the two-pass deduplication	≥ 95%	97% (10% sample, ~30K records)

Commented [NT20]: En esta parte me confundo si el contenido de la columna se mueve a la deracha o sisigue para abajo...de nuevo entiendo que deba ir para abajo y todas estas tablas deben ir en un apendice...

Commented [JV21R20]: Addressed & Implemented

Limitations

One limitation bounds the empirical claims: the reported KPIs (Table 5) draw from two samples of different scales. Pipeline latency and deduplication rate were measured on a stratified 10% sample (~30,000 records), while NIGP QA precision and Generate Product Name compliance were measured on a 50-item manual subset because sampling-based human review at the 10% scale is not tractable under the current scope.

Conclusions

The General Services Administration of Puerto Rico's procurement catalog has suffered a structural fragmentation problem severe enough to make auditability, spend analytics, and AI-driven decision-making (all called for by the government-wide strategy) unreliable. This paper has presented the design and implementation of an agentic Big Data platform that addresses fragmentation through a Databricks Medallion lakehouse and Microsoft Foundry agents, publishing a governed Item Master as timestamped CSV and NDJSON extracts for downstream consumption.

Four classes of artifact have been delivered against the four implementation objectives in the Problem section: an approved item identifier with simplified procurement variants; a daily Bronze ingestion job validated for the Silver and Gold transformations; five Foundry agents bounded by responsible-AI safeguards; and a three-dimensional governance framework (platform, process, operational). The result is a reusable Lakehouse-plus-agents pattern for public-sector catalog modernization. All four pipeline targets were met on the measured samples (Table 5); full-scale measurement of the two human-review KPIs is left for future work. The as-deployed first-year cost (~$72,000 USD) compares favorably with the original $200,000–$1,000,000 USD hand-curation quotes (Table 4).

Future Work

Two directions extend this work beyond the June 2026 handover. First, a Catalog Steward review-and-reinjection workflow over nigp_quarantine (the low-confidence NIGP holding table), closing the loop on low-confidence agent outputs through audit-logged corrections; the architectural hooks are in place, but the steward-facing tooling is out of scope. Second, a requisition-matching mode (NIGP QA Mode 2) that scores incoming requisition lines against item_master_gold and routes low-confidence matches to the same quarantine path.

Commented [NT22]: Significado??

Commented [JV23R22]: Nigp_quarantine es una tabla delta que existe para guardar los records que no pasan la prueba de confianza de los agentes.

Commented [NT24]: Hay que poner coma (,)

Commented [JV25R24]: Addressed & Implemented

Acknowledgements

I thank my advisor, Nelliud D. Torres Batista, for technical guidance, and the Polytechnic University of Puerto Rico Graduate School. At ASG, I thank Karla Mercado (Administrator and CPO), Joel Fontánez González (Sub-Administrator) and James Olmeda Rivera (CIO) for the institutional

support that enabled this engagement and the funding of this project.

AI USE DISCLOSURE

This work used Claude (Anthropic) for grammar, structure, and stylistic editing. All interpretations, arguments, and conclusions are the author's own and have been verified for accuracy.

REFERENCES

[1] WIPR. (2025, July 17). ASG anuncia solicitud de propuestas para integrar inteligencia artificial en el gobierno. [Online]. Available: https://wipr.pr/asg-anuncia-solicitud-de-propuestas-para-integrar-inteligencia-artificial-en-el-gobierno/.

[2] Estado Libre Asociado de Puerto Rico. (2019). *Ley Núm. 73 de 2019 — Ley de la Administración de Servicios Generales para la Centralización de las Compras del Gobierno de Puerto Rico* [Online]. Available: https://bvirtualogp.pr.gov/ogp/Bvirtual/LeyesOrganicas/pdf/73-2019.pdf.

[3] Administración de Servicios Generales de Puerto Rico. (2021, Julio 23). *Enmiendas al Reglamento Uniforme de Compras y Subastas (Reglamento Núm. 9292)* [Online]. Available: https://asgwebpageprodstorage.azureedge.net/media/reglamentos/05._Enmienda_a_Reglamento_de_Compras_9292.pdf.

[4] Social Glass, Inc., "*ASG Item Master List — Implementation Framework*," Social Glass, Inc., San Juan, PR, Internal Tech. Rep. Deliverable 2, 2025.

[5] Social Glass, Inc., "*ASG Item Master List — Diagnostic and Problem Definition*," Social Glass, Inc., San Juan, PR, Internal Tech. Rep. Deliverable 1, 2025.

[6] OECD, Governing with Artificial Intelligence: The State of Play and Way Forward in Core Government Functions. Paris, France: OECD Publishing, 2025. Doi: 10.1787/795de142-en.

[7] S. Lababidi, "A Critical Evaluation of ChatGPT's Adherence to Responsible Artificial Intelligence Principles," Information Sciences with Applications, vol. 3, 2024. Doi: 10.61356/j.iswa.2024.3303.

[8] A. K. Elmagarmid et al., "Duplicate record detection: A survey," IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 1, pp. 1–16, 2007.

[9] Microsoft. (n. d.). *What is the Medallion Lakehouse Architecture?* [Online]. Available: https://learn.microsoft.com/en-us/azure/databricks/lakehouse/medallion. [Accessed: Sept. 22, 2025].

[10] Microsoft Learn [Online]. Available: https://learn.microsoft.com/en-us/azure/databricks/. [Accessed: Sept. 22, 2025].

[11] Microsoft. (2026). *What is Microsoft Foundry Agent Service? Microsoft Learn* [Online]. Available: https://learn.microsoft.com/en-us/azure/foundry/agents/overview. [Accessed: Sept. 22, 2025].

[12] News Is My Business. (2025, May 23). GSA launches Mercadito, 1st gov't e-commerce platform in Latin America [Online]. Available: https://newsismybusiness.com/gsa-launches-mercadito-1st-govt-e-commerce-platform-in-latam/

[13] Administración de Servicios Generales de Puerto Rico. (n.d.). *Contract #2026-000049 (Social Glass, Inc.), ASG Transparency Portal* [Online]. Available: https://web.archive.org/web/20260510052653/https://transparencia-frontend.icyfield-893fab1f.eastus.azurecontainerapps.io/contratos/151612. [Accessed: May 9, 2026].