Stakeholder Requirements and Source System Understanding

Purpose of This Document

Although this is a portfolio project based on fictional data and simulated context, it closely mirrors how real-world analytics projects are initiated by capturing stakeholder requirements and understanding the technical landscape of the source systems involved.

This document captures how stakeholder needs and source system realities are jointly assessed to design a scalable full-stack Business Intelligence solution (from Data Lakehouse in Databricks to Power BI reports) for a retail organization with fragmented reporting. Rather than starting with tools or dashboards, the focus should be on:

- Understanding decision-making gaps
- Clarifying business questions
- Translating those needs into actionable analytics and BI requirements.

Strategic Context

The organization operates with multiple operational systems:

- An ERP system managing product and transactional data
- A CRM system managing customer attributes and engagement details.

Due to this fragmented setup:

- KPIs were calculated differently across teams
- Reporting required heavy manual effort
- Leadership lacked a unified view of revenue performance, customer behavior, and product profitability.

The BI initiative aimed to move the organization from reporting inconsistency to decision alignment.

Requirements Discovery Approach

In real-world scenarios, designing a scalable data lakehouse begins with two parallel tracks:

- **Stakeholder Collaboration**: Framing business questions and converting stakeholder needs into user stories to define KPIs, reporting expectations, and pain points in existing workflows. In short, this track focuses on *what decisions needed to be supported* and *why*.

- **Source System Analysis**: Understanding the structure, limitations, and integration options of the systems holding the raw data (e.g., ERP, CRM). In short, this track focuses on *what data exists, where it lives, and how reliable it is*.

To simulate this process, the project used a list of best-practice discovery questions that would typically be explored in stakeholder workshops and technical walkthroughs. These questions shape everything from ingestion and transformation logic to the architecture of reporting layers.

Discovery questions Used to Drive Requirements

- Who owns the data in each source system (ERP, CRM, etc.)?
- What business processes do these systems support (e.g., Sales, Customer Retention)?
- What data formats and storage mechanisms are in place (CSV, SQL Server, Oracle, cloud storage)?
- What are the integration capabilities? (API, file extracts, direct DB access, Kafka, etc.)
- What are the peak load times or usage periods for these systems?
- What is the data refresh frequency (daily, hourly, or real-time)?
- Should we implement full loads or delta (incremental) loads?
- How large are the typical data extracts? Are there any volume constraints?
- Are there known data quality, consistency, or completeness issues?
- Which fields are business-critical for KPIs and reporting?
- How will we validate data correctness post-ingestion?
- What level of historization (Type 1 vs. Type 2) is required?
- What are the reporting pain points that this data lakehouse is expected to solve?

Together, these stakeholder-driven stories and system-level insights informed the design of the data lakehouse using Medallion Architecture, ensuring technical feasibility and business relevance were aligned from the very beginning.

Sample User Stories & Acceptance Criteria

The user stories shown below represent a curated subset of the broader backlog (10+ stories) created during requirements discovery. These examples illustrate how stakeholder questions were translated into BI enablement, analytics, and reporting requirements.

User Story 1 - Executive Performance Alignment

As a COO, I want a single, consistent source of truth for sales, customers, and products, so that leadership decisions are based on reliable, reconciled data rather than disconnected reports.

Acceptance Criteria:

- Integrated ERP and CRM data in a centralized lakehouse
- Standardized definitions for Revenue, Orders, AOV, and Profit
- Data refreshes are automated (pipeline orchestration) and auditable
- KPIs are consistent across analytics and reporting layers

User Story 2 - BI Enablement & Scalability

As a BI Team, we want standardized, business-ready data so that analytics and reporting can scale without repeated manual effort.

Acceptance Criteria:

- Data is cleansed and standardized before reporting
- KPIs are defined once and reused
- Reporting tools connect to governed datasets.

User Story 3 - Analytics & Business Insights

As a BI Analyst, I want to analyze trends in order volume, customer behavior, and pricing effectiveness so that I can explain why performance changed, not just what changed.

Acceptance Criteria:

- Ability to analyze trends directly in SQL
- Support for customer segmentation and product profitability analysis
- Metrics can be sliced by time, customer type, and product category
- Outputs can be reused for both analysis and reporting.