Business Requirement Document (BRD)

# Project Title:

Post-Merger Customer Data Deduplication Framework

# 1. Objective

To develop a robust and scalable framework to identify, match, and merge duplicate customer records originating from two merged organizational databases. The outcome is a unified, de-duplicated master customer dataset.

# 2. Scope

This framework will handle ingestion, matching (exact and fuzzy), merging, and storage of customer data from two input CSV files into a master customer dataset. It also includes generating reports of matched records.

# 3. Input Data

Two CSV files representing customer records from two organizations post-merger. Both files share the same schema with fields like: CustomerID, First Name, Last Name, Phone Number, Address Line 1, City, State, Zip Code, Email.

# 4. Matching Criteria and Logic

Primary Match Key:  
- Match by exact Phone Number.

Secondary Match Key:  
- If phone is unavailable or does not match, match by Address.

Fuzzy Address Matching:  
- Use fuzzy matching (token sort ratio) on address fields.  
- Consider addresses with 80%+ similarity as matches.

# 5. Merge Conflict Resolution Strategy

- Prefer non-null fields.  
- Prefer most recently updated record.  
- Combine non-conflicting data (e.g., merge multiple emails).  
- Maintain source metadata for audit.

# 6. Technology Stack

- SQL: For initial transformations and storing results.  
- Python: For fuzzy matching and data logic.  
- PySpark: For handling large datasets and scalable processing.

# 7. Outputs

- Final de-duplicated master customer dataset.  
- Match report detailing matched records and fields.  
- Stored in a SQL-based table or data lake.

# 8. Performance and Scalability

- Use PySpark for distributed processing.  
- Optimize joins (broadcast, partitioning).  
- Use vectorized UDFs for fuzzy logic.  
- Filter comparisons by blocking (e.g., ZIP code groups).