Software Requirements Specification (SRS)

AI-Powered Merchant Data Cleaning Desktop Application

Version: 1.1 (Updated with Column Mapping and Row Range Features)
Date: August 31, 2025
Previous Version: 1.0 (August 20, 2025)

1. Introduction

1.1 Purpose

To provide a precise requirements specification for an AI-driven Windows desktop application that automates merchant data cleaning for large-scale, credit card merchant datasets, optimizing for usability, cost, and operational flexibility with enhanced user control over data processing.

1.2 Intended Audience

- Client company management and technical staff
- Software architects & development teams
- QA and support teams
- Future operational users (supervisors, reviewers)

1.3 Scope

The application will:

- Clean and standardize unstructured merchant strings from Excel files
- Enrich records with website/social media, verified business info, and evidence
- Minimize manual internet research with AI automation
- Enable easy configuration and operation by non-technical personnel
- **NEW**: Provide flexible user-controlled column mapping and row range selection
- Optimize for both cost and data accuracy
- Scale to process millions of records in batch mode with robust recovery

2. Overall Description

2.1 Product Perspective

The system is a standalone Windows desktop GUI application (Tkinter), with no server/backend required. All API interactions—Google Gemini Flash, Google Search, and optionally Google Places—are managed locally. No integration into client cloud, email or AD infrastructure is required.

2.2 User Needs

- Fast, easy onboarding for non-technical users (no hidden files or terminal use)
- Complete processing of Excel files with built-in cost and budget management
- **NEW**: Full control over which columns contain what data and which outputs go where
- **NEW**: Ability to process specific row ranges rather than entire files
- Clear batch job monitoring, recovery, and checkpointing, even for multi-day runs
- Easy review and manual follow-up on AI decisions

2.3 Assumptions and Dependencies

- Internet connection required for API usage
- Valid/fresh API keys must be provided via app GUI
- Users split large jobs into recommended batch sizes (≤200K)
- Output is passed to human logo collection and checking teams post-processing
- **NEW**: Users understand their data structure for effective column mapping

3. System Features & Functional Requirements

3.1 File Input/Output & Data Model

- **FR1**: Accept Excel files (.xlsx, .xls) with up to 3M rows, supporting batch chunking
- **FR2**: ~~Flexible header mapping – auto-detect variants of merchant name/address fields~~ **[SUPERSEDED BY FR2A-FR2E]**

- **FR2A**: **[NEW]** Display interactive Column Mapping GUI after file selection showing:
    - Preview of first 10 rows of data
    - Auto-detected column suggestions as starting point
    - Dropdown menus for each column to map to required fields
- **FR2B**: **[NEW]** Required field mappings:
    - Merchant Name (mandatory)
    - Address/Street (optional)
    - City (optional)
    - Country (optional)
    - State/Region (optional)
- **FR2C**: **[NEW]** Validation rules for column mapping:
    - Prevent multiple input columns mapping to same output field
    - Show clear error message if duplicate mapping attempted
    - Highlight conflicts in red with specific resolution instructions
- **FR2D**: **[NEW]** Save/Load column mapping configurations:
    - Save mappings as .json files with user-defined names
    - Load previously saved mappings for similar file formats
    - Default mapping library for common merchant data formats
- **FR2E**: **[NEW]** Column mapping confirmation screen showing:
    - Summary table of all mappings
    - Sample data preview with mapped fields highlighted
    - Edit/modify option before proceeding to processing
- **FR3**: Output must match input format, preserving order and non-standard columns
- **FR4**: Add required new columns: Cleaned Merchant Name, Website, Social(s), Evidence, Evidence Links, Cost per row, Logo Filename, Remarks

3.2 Row Range Selection & Processing Control

- **FR5A**: **[NEW]** Row Range Selection Interface:

    - Input fields for "Start Row" and "End Row" numbers
    - Default to full file range (Row 2 to last row, assuming headers in Row 1)
    - Live validation ensuring Start Row ≤ End Row
    - Display total rows selected and estimated processing time
- **FR5B**: **[NEW]** Row Range Preview:
    - Show preview of first 5 and last 5 rows in selected range
    - Display total record count for selected range
    - Cost estimation based on selected rows only
- **FR5C**: **[NEW]** Range Processing Integration:
    - All processing (cleaning, enrichment, cost calculation) applies only to selected range
    - Checkpoint/recovery works within selected range boundaries
    - Output file contains only processed rows plus original headers
- **FR5D**: **[NEW]** Multiple Range Processing:
    - Support for processing multiple non-consecutive ranges in separate jobs
    - Save range selections for reuse with similar files
    - Warning if ranges overlap or leave gaps

3.3 Operation Modes

- **FR6**: Two processing modes selectable in-GUI:
    - Basic: Google Gemini Flash AI + Search API
    - Enhanced: Adds Google Places API for business verification
    - Users can toggle modes on job setup screen
    - Mode directly impacts accuracy and cost

3.4 API Key Management & Configuration

- **FR7**: All API key setup is via GUI dialog on first launch or reconfiguration
- **FR8**: API keys (Gemini, Search; Places optional) stored locally in protected config file (option for encryption)
- **FR9**: No .env file requirement; no email/SNMP/SMTP dependencies

- **FR10**: Users can modify/update/replace keys in-app at any time
- **FR11**: Local config is not stored in version control and can be included in backup/recovery archives

3.5 Cost Control & Budget Enforcement

- **FR12**: Pre-processing cost estimation for every batch, showing:
  - Estimated cost per row
  - **NEW**: Cost calculation based on selected row range only
  - Probability of staying below ₹3/row (shows for both Basic/Enhanced)
- **FR13**: GUI confirmation and warning before any run likely to exceed budget
- **FR14**: Hard per-row budget controller: abort if run exceeds budget limit
- **FR15**: Row-level and session-level cost tracking (output column and summary)

3.6 Cleaning Workflow & Core Logic

- **FR16**: 6-step search logic, stopping on success:
  a. Merchant + address + city + country
  b. Merchant + city + country
  c. Merchant + city
  d. Merchant + country
  e. Merchant only
  f. Merchant + street
- **FR17**: AI-powered name cleaning (Gemini)
- **FR18**: Website/social enrichment via Search, Places (Enhanced), and social API/scraping if available
- **FR19**: Robust fallback when Places API disabled
- **FR20**: Evidence column must include structured explanation (as per agreed template)
- **FR21**: Logo naming rules handled in-app; logo collection still manual
- **FR22**: Each row's full processing cost written to output

3.7 Edge Case & Business Logic Handling

- **FR23**: Optional Places API is made explicit in all documentation and the GUI; fallback logic is robust if Places is not available

- **FR24**: AI must follow exact validation/acceptance business rules as previously specified (including for similar business names, closed listings, franchise name cleaning, aggregator-only listings, social media/website criteria)
- **FR25**: If no valid match after all queries, log as "NA" in remarks
- **FR26**: Evidence and decision trail must be detailed, explaining fallback and matching if needed
- **FR27**: **[NEW]** Column mapping error handling:
    - Graceful handling of missing required columns
    - Clear error messages for unmappable data types
    - Recovery options when mapping fails during processing

3.8 Notifications & User Feedback

- **FR28**: All process alerts (start, finish, error, batch over budget, completion) use Windows/native popups or app modal dialogs
- **FR29**: No email, SNMP, or 3rd-party push alerts; robust log file for troubleshooting

3.9 Job Management and Recovery

- **FR30**: Batch jobs can be paused/resumed/stopped by user from GUI
- **FR31**: Checkpoint/recovery every 50–100 rows for long jobs; user can restart from last checkpoint
- **FR32**: **[NEW]** Checkpoints respect row range selections and column mappings
- **FR33**: Checkpoints portable between machines with same app version

3.10 Security and Data Protection

- **FR34**: All config and result files must be protected by local OS permissions
- **FR35**: Encryption of config/logs optional but recommended for corporate/PII deployments
- **FR36**: No sensitive data stored in cleartext in version control
- **FR37**: User manual includes backup and recovery steps for config/checkpoint data

3.11 Testing & Documentation

- **FR38**: All features to be tested in both Basic and Enhanced (Places) modes
- **FR39**: **[NEW]** Additional tests for:

    - Column mapping validation and conflict detection
    - Row range selection and processing accuracy
    - Mapping configuration save/load functionality
    - Cost estimation accuracy for selected ranges
- **FR40**: Tests for key entry/rotation, job resume, cost control, evidence formatting, and checkpoint file migration
- **FR41**: Documentation to match GUI-first usage (no hidden files/cmdline)

4. Non-Functional Requirements

4.1 Performance

- Capable of processing at least 500–1000 rows/hour per instance, given API rate constraints
- Memory use stays <4GB due to chunked streaming and SQLite caching
- **NEW**: Column mapping and row selection operations complete within 2 seconds for files up to 1M rows

4.2 Portability & Upgrades

- App, config, and checkpoint data fully portable
- **NEW**: Column mapping configurations portable across machines
- Easy local backup/restore
- Direct migration from any old .env-based or SMTP-dependent version included in docs

4.3 Usability

- No actions require use of command line, scripting, or file edits outside the app GUI
- All errors and confirmations communicated in plain, actionable messages
- **NEW**: Intuitive drag-and-drop or point-and-click column mapping interface
- **NEW**: Visual validation of all user selections before processing begins

4.4 Cost Management

- Cost per row displayed on batch estimate and output

- Hard stop on cost > ₹3.00/row
- Users informed of trade-offs between Basic and Enhanced modes
- **NEW**: Real-time cost updates as row ranges are modified

4.5 Disaster Recovery

- Checkpointing ensures near-zero lost work on crash/interruption
- Recovery instructions included with each backup/export
- **NEW**: Column mapping and range selections preserved in checkpoint files

4.6 Security

- API key config files with permissions 600 or higher by default
- PII handling matches organizational policy; encryption toggle in advanced options

5. User Interface Workflow

5.1 New Enhanced Workflow

1. **File Selection**: User selects Excel file
2. **NEW: Column Mapping Setup**:
    - Auto-detection displays suggested mappings
    - User reviews and modifies mappings via dropdown menus
    - System validates no duplicate mappings exist
    - User can save mapping configuration for future use
3. **NEW: Row Range Selection**:
    - User specifies start and end rows
    - System shows preview and count of selected rows
    - Cost estimation updates based on selected range
4. **Processing Mode Selection**: Basic or Enhanced
5. **Final Confirmation**: Review all settings before starting

6. **Processing & Monitoring**: Standard processing with enhanced checkpointing
7. **Results Review**: Output file contains only processed rows with all specified columns

5.2 Column Mapping GUI Components

- **File Preview Panel**: Shows first 10 rows with column headers
- **Mapping Configuration Panel**:
    - Source column dropdown for each required field
    - Visual indicators for mapped/unmapped columns
    - Error highlighting for conflicts
- **Validation Panel**: Real-time validation messages
- **Save/Load Panel**: Manage mapping configurations

5.3 Row Range Selection GUI Components

- **Range Input Panel**: Start/End row number inputs with validation
- **Preview Panel**: Shows sample of selected data
- **Summary Panel**: Total rows, estimated processing time, estimated cost

6. Appendix

6.1 Modes:

- **Basic (Default)**:
    - Gemini Flash + Search API
    - Lower cost, basic web verification
    - All business logic rules followed
- **Enhanced**:
    - Adds Google Places API for higher confidence matching
    - Higher cost, improved accuracy
    - Optional; user decides per batch

6.2 Change Log/Traceability

Version 1.1 Changes (August 31, 2025):

- Added comprehensive column mapping functionality (FR2A-FR2E, FR27)
- Added row range selection and processing (FR5A-FR5D)
- Enhanced cost estimation for selected ranges (FR12)
- Added checkpoint integration with new features (FR32)
- Updated testing requirements for new features (FR39)
- Enhanced UI workflow specification
- Added performance requirements for new features

Version 1.0 (August 20, 2025):

- Initial release with all core functionality
- This SRS is the authoritative source for all design, implementation, and test planning
- Any reference to .env, SMTP/email, or mandatory Places API in docs/code is superseded by GUI-based configuration, native notifications, and two-tier verification workflow (Basic/Enhanced)

All requirements above are mandatory for v1.1 delivery.

End of SRS v1.1 – August 31, 2025