

Spec Doc: Manual Claims Processing (Single & Bulk)

Author	Athman Gude
Published	Jan 20, 2026
Parent	Streamlining Jireh Payments
Reviewers	Terach F., David N., Duncan G., Azadi Gathui
Approvers	Terach F.
Status	In Review

1. Overview

This module facilitates the manual submission of claims by providers who do not have direct API integrations. To reduce friction and data entry errors, the system utilizes a **Local-First Processing** approach. Users upload files (PDF, CSV, Excel), and the browser processes them locally to extract data *before* any data is sent to the server. This ensures data privacy, speed, and allows the user to verify extraction accuracy immediately.

2. Core Features

2.1. File Agnosticism & Local Extraction

- **Capability:** The system accepts various formats for both single and bulk claims, including PDF, JPG, PNG, CSV, and XLSX. This allows providers to upload multi-page PDFs or images of invoice lists for bulk processing, not just structured spreadsheets.
- **Local Logic:** Instead of uploading the raw file to Jireh's servers immediately, a browser-based parser (OCR for images/PDFs, Parsing for Spreadsheets) extracts the critical billing data. Crucially, this includes **Invoice Items** (line-item details of services/medication provided), alongside **Invoice Number** and **Amount**. *Note: Patient Name is not a mandatory extraction field, but Item details are required for auditing.*
- **Intelligent Auto-Mapping:** The system utilizes smart heuristics to automatically detect column headers and data structures in bulk files. Manual column mapping is presented only as a **last resort** if the system cannot confidently identify the required fields (Invoice #, Amount, and Items).

2.2. The Verification Table (Staging Area)

Before submission, data sits in a "Staging Table."

- **Editable:** Users can correct OCR errors (e.g., if "500" was read as "5000", or "Paracetamol" was read as "Parasail").
- **Reactive Calculations:** Once corrections are made, the system locally updates any arithmetic pertaining to totals (such as "Total Batch Value") with the corrected values immediately, ensuring the summary reflects the user's edits before API verification.
- **Validation:** Users trigger a "Verify against Jireh" action. This sends *only* the data (not the file) to the API to check validity.

2.3. Claims Dashboard

The Claims Dashboard acts as a unified command center for providers to track and reconcile their financial history. Rather than a static list, it offers a dynamic table view where every submission is tracked by its date, invoice identifier, and total value. The status of each claim is clearly visible, allowing users to instantly distinguish between items that are processing, fully settled, or rejected. Direct actions are available to **View Details** (which expands to show the specific **Invoice Items** captured), or download remittance advice. To facilitate easy management of large datasets, the dashboard is equipped with robust filtering tools. Providers can drill down to view only unsettled claims for immediate action, filter history by specific date ranges such as the last 30 days, or use the search function to pinpoint specific invoice numbers, ensuring that no payment ever slips through the cracks.

3. User Flows

3.1. Flow A: Single Claim (Small Provider)

Ideal for: Individual doctors or small clinics processing one patient at a time.

For individual practitioners and smaller clinics, the single claim workflow begins simply with a drag-and-drop action of the invoice file—whether it be a PDF or an image—directly into the browser. Immediately upon upload, the system initiates a local scanning process, utilizing browser-based OCR to extract key data points without sending the file to the server. The user is then presented with a review modal that places the original invoice image side-by-side with the extracted data fields. **Crucially, this view lists the extracted 'Invoice Items' (e.g., 'Consultation Fee', 'Lab Test')**, allowing the provider to verify that the service details were captured correctly alongside the invoice number and total amount. This interface allows for immediate correction of any extraction errors. Once satisfied, the user triggers a verification check against the Jireh API. The system provides instant feedback: a green checkmark indicates a match with internal records, while an error message highlights discrepancies like amount mismatches. Only after this validation step does the user confirm the submission, saving the structured data as a verified claim.

3.2. Flow B: Bulk Claim (Large Provider)

Ideal for: Hospitals or networks processing weekly/monthly batches.

Designed for hospitals and larger networks managing high volumes, the bulk claim workflow accepts multi-row files in formats like CSV, Excel, or batched PDFs. Upon upload, the system attempts to auto-detect the file structure to identify headers and rows; if the format is unrecognized, it prompts the user to map columns (e.g., identifying which column contains the Invoice Number, Amount, and **Invoice Items/Description**) as a last resort. The data is then populated into a local staging table grid, where **Invoice Items** are visible and editable. The user initiates a batch verification, sending the data to the API for validation. The interface visually segments the results: valid rows where the invoice number and amount match Jireh records turn green, rows with unknown invoice numbers turn red, and rows with matching numbers but conflicting amounts turn yellow. The user can then resolve these exceptions by editing values to fix typos, deleting invalid rows, or choosing to proceed by submitting only the valid green rows.

4. Validation Logic (API Interaction)

The verification step checks data points against the Jireh Backend and ensures data integrity:

State	Invoice Number	Amount	Invoice Items	UI Feedback	Action
Valid	Found	Matches	Present	Green Check	Ready to Submit
Amount Mismatch	Found	Mismatch	Present	Yellow Warning: "Jireh expects KES 2,300"	User must correct amount or contest
Missing Items	Found	Matches	Empty/Null	Red Warning: "Missing Service Details"	User must enter what was billed
Unknown Invoice	Not Found	N/A	N/A	Red Error: "Invoice not found"	User must correct number or remove
Duplicate	Found	Matches	Present	Orange Warning: "Already claimed"	Block submission

5. Scenarios

5.1. Scenario: Single Invoice OCR Correction

Goal: Submit a single invoice where the OCR misreads the amount or item description.

Experience: Dr. Amina uploads a PDF invoice. The system extracts the amount as "50.00" instead of "5000" and misreads "X-Ray" as "X-Rav". She sees these errors in the side-by-side preview. She manually corrects the amount to "5000" and the item description to "X-Ray". She clicks "Verify". The system confirms "Match found for 5000". She submits the claim.

5.2. Scenario: Bulk Upload with Mixed Results

Goal: Process a spreadsheet where some invoices match and others don't.

Experience: The Admin uploads March_Claims.xlsx (50 rows). The Staging Table loads, displaying columns for Invoice #, Amount, and Description. She clicks "Verify".

- 45 rows turn Green.
- 3 rows turn Red (Invoice numbers typed incorrectly in her Excel).
- 2 rows turn Yellow (Amounts are off by 1 shilling).

She deletes the 3 Red rows. She corrects the 2 Yellow rows to match the Jireh expected amount. She clicks "Submit 50 Claims".

5.3. Scenario: Column Mapping (Unknown Structure)

Goal: Upload a file with a non-standard format ensuring line items are captured.

Experience: Admin uploads a file where the invoice number is in a column named "Ref_ID" and services are under "Service_Desc". The system alerts: "We couldn't auto-detect columns." It displays a sample row and asks: "Which column is the Invoice Number?" (Selects Ref_ID), "Which column is the Amount?" (Selects Total), and "**Which column contains the Items?**" (Selects Service_Desc). The Admin confirms, and the system processes the file using this mapping.

6. User Stories

6.1. Flow A: Single Claim (Small Provider)

Title	Description	Acceptance Criteria
Local OCR Extraction	As a Small Provider, I want my invoice to be scanned locally in my browser, so that my sensitive files are not uploaded until I am ready to claim.	<ol style="list-style-type: none">1. System accepts PDF, JPG, PNG via drag-and-drop.2. OCR extracts Invoice #, Amount, and line items (Description).3. Extraction happens locally without network calls for the file.
Side-by-	As a Provider, I want to	<ol style="list-style-type: none">1. Modal displays original document image on the

Side Review	see a side-by-side view of the invoice image and extracted fields, so that I can easily verify accuracy.	<p>left.</p> <p>2. Editable fields (Invoice #, Amount, Items) appear on the right.</p> <p>3. Line items are displayed as a list or table for clarity.</p>
Manual Correction & Recalculation	As a Provider, I want to manually edit any misread data points, so that the claim submission is accurate.	<p>1. All extracted fields are editable.</p> <p>2. Changing line item amounts updates the "Total Amount" field automatically.</p> <p>3. System prevents submission of empty mandatory fields.</p>
Instant API Verification	As a Provider, I want to verify my invoice data against Jireh's records, so that I know the claim will be accepted.	<p>1. "Verify" button triggers an API check for Invoice # and Amount.</p> <p>2. Visual feedback (Green/Yellow/Red) is displayed based on validation logic.</p> <p>3. Error messages specifically state if the amount mismatches Jireh's expected value.</p>
Single Claim Submission	As a Provider, I want to finalize my verified claim, so that I can receive payment.	<p>1. Submission button is enabled only after verification.</p> <p>2. Successful submission redirects to the Claims Dashboard.</p> <p>3. Claim status is immediately updated to "Processing".</p>

6.2. Flow B: Bulk Claim (Large Provider)

Title	Description	Acceptance Criteria
Bulk Batch Upload	As a Large Provider, I want to upload multiple claims in a single file	<p>1. Supports multi-record spreadsheets and multi-page PDFs.</p>

	(CSV/Excel/PDF), so that I can process my weekly volume efficiently.	<p>2. System indicates upload progress and extraction status.</p> <p>3. Total batch count and value are displayed after extraction.</p>
Intelligent Auto-Mapping	As an Admin, I want the system to automatically recognize my file structure, so that I don't waste time on manual column mapping.	<p>1. System uses headers like "Inv", "Amount", "Description" to auto-map.</p> <p>2. Successfully mapped records are moved directly to the staging table.</p> <p>3. A confirmation summary shows which columns were identified.</p>
Mapping Last Resort	As an Admin, I want to manually map columns if auto-detect fails, so that I can process non-standard report exports.	<p>1. Mapping interface appears only when auto-detection confidence is low.</p> <p>2. User selects columns for Invoice #, Total, and Items from a dropdown.</p> <p>3. System saves mapping preference for future files from the same provider.</p>
Bulk Staging Management	As an Admin, I want to manage all extracted claims in a grid, so that I can perform bulk edits or deletions.	<p>1. Staging table allows sorting and searching by Invoice #.</p> <p>2. Inline editing is enabled for Amount and Description/Items.</p> <p>3. Mass delete option is available for invalid rows.</p>
Segmented Batch Verification	As an Admin, I want to verify the entire batch and see color-coded results, so that I can focus on resolving errors.	<p>1. "Verify All" button triggers batch API validation.</p> <p>2. Green rows = Perfect match.</p> <p>3. Yellow rows = Mismatching amount (displays Jireh's expected amount).</p> <p>4. Red rows = Invoice # not found.</p>
Selective Bulk	As an Admin, I want to submit only the valid	<p>1. "Submit Valid Rows" button is available.</p>

Submission	rows from a batch, so that my payments are not delayed by a few problematic invoices.	2. Valid rows are submitted to API; invalid rows remain in the staging table for correction. 3. System provides a summary of successfully submitted vs. held claims.
Audit Detail Visibility	As an Auditor, I want to see the specific items included in each bulk claim, so that I can verify the legitimacy of the billing.	1. Detailed "Items" column is present in the staging table. 2. Clicking an item list expands it to show full descriptions and unit prices (if available). 3. Item presence is a mandatory condition for "Green" status.