**User Story: US 5.1 – Import and Export Items via Excel for Opening Balance Invoice**

**As** an inventory accountant **I want** to import and export items to and from the Opening Balance Invoice using an Excel file **So that** I can quickly bulk-manage inventory data with accurate validations and flexible review/editing options

**Acceptance Criteria**

**A. Interface Features**

1. **"Import from Excel" button** allows uploading items to the Opening Balance Invoice.
2. **"Download Template" button** downloads a preformatted Excel template with required columns.
3. **"Export to Excel" button** allows exporting the current invoice items to Excel.
4. **Each item row** in the UI has an Edit button that allows modifying only that specific row.

**Excel Template Structure**

The default import file should have the following **columns** (in the same order):

|  |  |  |  |
| --- | --- | --- | --- |
| **Column Header** | **Arabic Label** | **Required** | **Notes** |
| Item Code | رقم الصنف | ✅ | Must exist in system or be added after confirmation |
| Item Name | اسم الصنف | ✅ | Display purpose, not used for match logic |
| Batch Number | رقم الباتش | Optional | Required only for **batch items** |
| Expiry Date | تاريخ الصلاحية | Optional | Required only for **batch items** |
| Item Attributes | سمات الصنف | Optional | Required only for **attribute-based items** |
| Unit | وحدة الصنف | ✅ | Must match one of the units linked to the item |
| Warehouse | المستودع | ✅ | Must be valid & linked to the current branch |
| Quantity | كمية الصنف | ✅ | Must be > 0 |
| Price | سعر الصنف | ✅ | Must be > 0 |

**Note:** There are 3 item types:

* عادي (Normal): No batch or attributes
* باتش (Batch-based): Requires Batch Number & Expiry Date
* له سمات (Attribute-based): Requires attributes field filled

**Functional Workflow**

**1. Upload Excel File**

* User uploads an .xlsx file via the "Import from Excel" button.

**2. Backend Processing (Validation Phase)**

* File is validated before anything is saved.

**Warehouse Validation**

* Warehouse must exist and be linked to the current branch.

**Item Validation**

* Items must exist in the system.
  + If not found in the warehouse, show a warning with list:

“These items are not found in the warehouse. Do you want to add them?”

* + - If **Yes**: items are added.
    - If **No**: only existing items in the warehouse are added.

**Attribute Validation**

* If the item has attributes in the Excel file:
  + Those attributes must already be linked to the item in the system.
  + If not → show alert:

“The following attributes are not linked to the item. Please link them first.”

* + - Items with unmatched attributes will **not be saved**.

**Unit Validation**

* If unit in Excel is not assigned to item → show:

“These items are not linked to the specified unit.”

* + Invalid rows are skipped.

**Quantity & Price Validation**

* Quantity and price must be greater than 0.
  + Else → skip row and show specific error.

**3. Display Validated Data to User**

* After backend sends back valid rows:
  + Show a table with all **valid items ready to save**.
  + Mark invalid/missing ones for **user review or export**.

**4. Item Editing Feature**

* Each row in the UI will have an Edit button:
  + Opens row for in-place editing.
  + Backend process:
    - Delete the item row from DB for this invoice.
    - Re-add updated data as new.

**5. Export Invalid Items**

* If any items were **rejected due to missing warehouse, unit, or attributes**, allow user to:
  + Click **"Export Invalid Rows"** to download the error list for offline correction.

**Excel Documentation Sheet Proposal**

To help users prepare/import correctly, structure the file as follows:

**Sheet 1: Template**

* First row: headers
* Second row: sample data for عادي / باتش / له سمات
* Protect structure with format rules

**Sheet 2: Validation Rules**

|  |  |
| --- | --- |
| **Field** | **Rule** |
| Warehouse | Must be linked to logged-in branch |
| Quantity | Must be > 0 |
| Price | Must be > 0 |
| Attributes | Required only for attribute-based items, must be linked to item |
| Unit | Must be one of the units linked to item |
| Expiry Date | Required for batch items only |

**Sheet 3: Rejected\_Items\_Log (auto-generated after import)**

* Includes reason for rejection per row

**How to Detect the Item Type**

|  |  |  |
| --- | --- | --- |
| **Type** | **Detection Logic (based on Excel input)** | **Required Fields** |
| **Normal Item** | No Batch Number, no Expiry Date, no Item Attributes | Item Code, Unit, Quantity, Price, Warehouse |
| **Batch Item** | Batch Number and Expiry Date are filled | + Batch Number, Expiry Date |
| **Attribute Item** | Item Attributes column is filled | + Item Attributes |
| **Batch + Attribute** | Both batch fields and attributes are filled | All of the above |

**Business Logic**

|  |  |
| --- | --- |
| **Condition** | **System Action** |
| Item not in warehouse | Warn user → Add or skip based on user response |
| Item has unmatched attributes | Block save → Ask user to fix and retry |
| Unit not linked to item | Skip row → Warn user |
| Quantity or price ≤ 0 | Skip row → Warn user |
| User edits an item row | Delete & re-insert item from backend |
| Unmatched/invalid rows found | Allow export to Excel for correction |