Take home exercise (Full Stack)

Note: Please don’t use AI based coding agent to complete this assignment. You can refer to internet for for specific help.

Key Definitions

Marketplace

A *marketplace* is an online platform where sellers list and sell their products.

Examples: **Myntra**, **Flipkart**, **Amazon**, etc.

Each marketplace has its own **product listing template** — a set of rules and attributes (like *product name*, *brand*, *price*, *size*, *color*) that every product must follow to be listed.

Think of it like a **form** you have to fill in:

1. Flipkart might ask for "Title", "MRP", "Listing Price", "Image 1", etc.
2. Myntra might ask for "Product Name", "Gender", "Category", "Selling Price", etc.

These templates vary slightly across marketplaces, even if they describe the same product.

Seller File (CSV/Excel)

A *seller file* is the data provided by the merchant (the person who wants to sell products).

It usually comes as a CSV or Excel file, with **product attributes** like:

1. SKU (unique code for each product)
2. Name (product title)
3. BrandName
4. Color, Size
5. Price, MRP
6. Images (links to product photos)
7. Quantity, Description

This file represents how the seller organizes their own product catalog — but its columns rarely match exactly with what marketplaces demand.

Attribute Mapping

Because each marketplace has different naming conventions and requirements, we need to **map** seller file columns → marketplace attributes.

Example:

- Seller file column Name → Myntra's productName
- Seller file column BrandName → Flipkart's brand
- Seller file column Image1 + Image2 → Myntra's images

The mapping step is **critical**: without it, the product data cannot be uploaded to the marketplace.

Assignment Goal (in simple words)

You'll build a **mini product-listing tool that**:

- **Uploads and save a marketplace template** (what attributes that marketplace expects).
- **Upload a products CSV/Excel** (what the seller actually has).
- **Map columns from the file → marketplace attributes** using a small UI.
- **Save the mapping to a database**
- **View the seller mapings**

What to build (end-to-end flow)

1. **Marketplace Template Upload**
    a. Frontend: Page to upload a **csv template**.
    b. Backend: API to **create/list/get** marketplace templates.
2. **Seller Product File Upload**
    a. Frontend: Page to upload **CSV**.
    b. Backend: API to parse and return **discovered columns/sample rows**.
3. **Mapping UI**
    a. Show two columns:
        i. **Source (seller product file columns)** — e.g., SKU, Name, BrandName…
        ii. **Target (marketplace attributes)** — e.g., productName, mrp, images…
4. Allow **one-to-one** mapping between product attributes and marketplace attributes.
5. **Save & Display Mapping**
    a. Save mapping to DB (marketplace + file + column map + timestamp).
    b. Show a **list page** of saved mappings.

You can use any stack (suggested: **React** for UI, **Node/Express** or **Python/FastAPI** for APIs, **SQLite/Postgres** for DB).

Marketplace Template and mapping example:

Note: You can use these examples to test your assignment. Create a csv of these example marketplace and seller attributes and upload, map and list mappings.

Myntra-like Marketplace Template

Required attributes

- productName (string, max 150 chars)
- brand (string)
- gender (enum: Men, Women, Boys, Girls, Unisex)
- category (enum: T-Shirts, Jeans, Dresses, Sarees, Shoes, Bags, Accessories)
- color (string)
- size (enum: XS, S, M, L, XL, XXL, numeric sizes like 32, 34, etc.)
- mrp (number ≥ 0)
- price (number ≥ 0, must be ≤ mrp)
- sku (unique string)
- images (array of URLs)
- description (string)
- material (string)

Flipkart-like Marketplace Template

Required attributes

- title (string, max 200 chars)
- brand (string)
- sellerSku (string)
- categoryPath (string, e.g., “Clothing > Men > T-Shirts”)
- listingPrice (number ≥ 0, must be ≤ mrp)
- mrp (number ≥ 0)
- color (string)
- image1 (URL)
- quantity (integer ≥ 0)
- size (string)

- gender (enum: Men, Women, Boys, Girls, Unisex)
- bulletPoints (array of up to 5 strings, | separated)
- image2, image3 (URL)
- description (string)
- countryOfOrigin (string, e.g., “India”)

Seller File Template (CSV/Excel)

Columns provided by the seller:

- SKU
- Name
- BrandName
- Gender
- Category
- Color
- Size
- MRP
- Price
- Material
- Image1
- Image2
- Quantity
- Description

Seller → Myntra Mapping

- productName ← Name
- brand ← BrandName
- gender ← Gender
- category ← Category
- color ← Color
- size ← Size
- mrp ← MRP
- price ← Price
- sku ← SKU
- description ← Description

- material ← Material
- images ← Image1

Deliverables

Note: The solution should run on all environments (windows, linux, macos):

Create a github repo and upload the below items:

- Source code (API + Frontend).
- README.md file that has:
    - Design, code and db documentation.
    - Setup and usage instructions.
    - Running unit tests instructions.

Evaluation Criteria

a. Correctness of mapping & validation.
b. Code quality (clean, modular, testable).
c. UI/UX clarity for mapping.
d. Extensibility (easy to add new marketplaces).
e. **Bonus points** for Docker based deployment and testing.
f. **Bonus points** for implemeting unit tests.