Software Requirement Document (SRD)

**Project Overview**

**Project Name:** Online Marketplace Vendor Management System Enhancement

**Client:** Online Market Place (OMP)

**Date:**

**Project Description**

The Online Marketplace Vendor Management System Enhancement aims to provide vendors with the ability to manage product listings, including product information, pricing, and availability, within an online marketplace. Vendors can have multiple catalogues, each with its own data format, and the system ensures data consistency by mapping vendor-specific data to a standardized schema.

**Architecture Overview**

**User Interface (UI)**

The UI includes the following components and features:

1. **Vendor Dashboard**: Provides access to vendor-specific data management. This page shows vendor all the available catalogue that has been updated by the vendor.
2. **Bulk Product Import/Export:** Provide ability to bulk import of products using CSV or Excel files
3. **Data Mapping Tool**: Allows vendors to map their data fields to a standardized schema.
4. **Product Management**: Enables vendors to add, edit, and delete product listings individually.
5. **Product Versioning:** Provide ability to add product versioning or variants by adding different versions of a product and verifying they are displayed correctly.
6. **Real-time Preview**: Offers real-time previews of product listings.
7. **Product Search and Filters:** Provide ability to search functionality with various search terms.
8. **Inventory Forecasting:** Display inventory forecasting tool that allows accurate forecasts about when and how much a given product is to be re-stocked.
9. **Multi-language Support:** Provide multi-language support by adding and viewing products in different languages.
10. **Inventory Alerts:** Provide inventory alert notifications for vendors when stock levels are low or products are out of stock.

**Backend Microservices**

The system comprises the following microservices:

1. Authentication and Authorization Service: Manages user accounts and permissions for vendors. Reused from the existing Vendor Management System. This reuses the same Authentication service from the Vendor Management module.
2. Data Ingestion and Transformation: Ingests data from vendors in various formats and transforms it into a standardized schema.
3. Product Management Service: Handles CRUD operations for products, including search, filtering, and validation.
4. Notification Service: Sends notifications to vendors about product updates.

**Data Store**

1. Relational Database: Stores structured data such as user information, product metadata, and order history.

Sample Data  
**Table: vendors**

|  |  |
| --- | --- |
| **Column Name** | **Data Type** |
| vendor\_id | VARCHAR(10) |
| vendor\_name | VARCHAR(255) |
| email | VARCHAR(255) |

|  |  |  |
| --- | --- | --- |
| **vendor\_id** | **vendor\_name** | **email** |
| "vendor\_001" | "Vendor A Inc." | "[vendorA@example.com](mailto:vendorA@example.com)" |
| "vendor\_002" | "Vendor B Ltd." | "[vendorB@example.com](mailto:vendorB@example.com)" |

**Table: catalogues**

|  |  |
| --- | --- |
| **Column Name** | **Data Type** |
| catalogue \_id | VARCHAR(10) |
| catalogue \_name | VARCHAR(255) |
| vendor\_id | VARCHAR(10) |

|  |  |  |
| --- | --- | --- |
| **catalogue \_id** | **catalogue \_name** | **vendor\_id** |
| "catalog\_001" | "Electronics" | "vendor\_001" |
| "catalog\_002" | "Clothing" | "vendor\_002" |

**Table: products**

|  |  |
| --- | --- |
| **Column Name** | **Data Type** |
| product\_id | VARCHAR(10) |
| name | VARCHAR(255) |
| description | TEXT |
| price | DECIMAL(10, 2) |
| availability | INT |
| catalogue \_id | VARCHAR(10) |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **product\_id** | **name** | **description** | **price** | **availability** | **catalogue \_id** |
| "101" | "Product X" | "This is Product X." | 49.99 | 100 | " catalogue \_001" |
| "102" | "Product Y" | "Description for Product Y" | 29.99 | 50 | " catalogue\_002" |

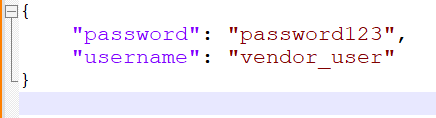
1. NoSQL Database: Stores unstructured or semi-structured data like product descriptions, images, and vendor-specific details.

**API Endpoints and Sample Data**

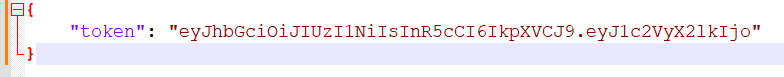
**Authentication and Authorization**

* **/api/auth/login**: POST request for vendor login.

**Sample Request Data:**



**Sample Response Data:**

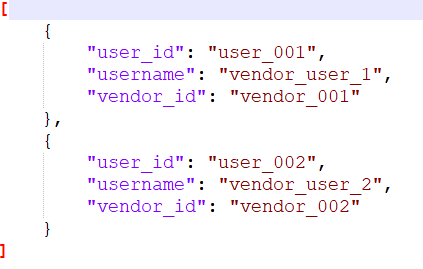


* **/api/auth/logout**: POST request for vendor logout.
* **/api/auth/check**: GET request to check authentication status.
* **/api/auth/reset-password**: POST request for password reset.
* **/api/auth/refresh-token**: POST request for token refresh.

**Admin APIs**

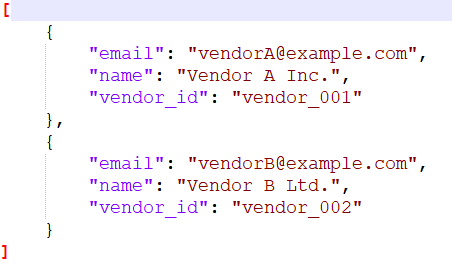
* **/api/users**: GET request to retrieve a list of all vendor users for administrative purposes.

**Sample Response Data:**



* **/api/vendors**: GET request to retrieve a list of registered vendors.

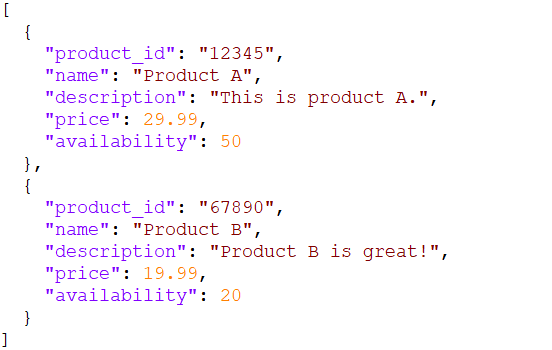
**Sample Response Data:**



**Vendor-Specific APIs**

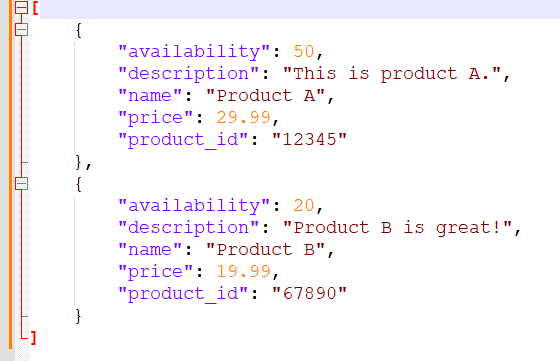
* **/api/vendors/{vendor\_id}/products**: GET request to retrieve products associated with a specific vendor.

**Sample Response Data:**



* **/api****/vendors/{vendor\_id}/orders**: GET request to retrieve order history for a specific vendor.

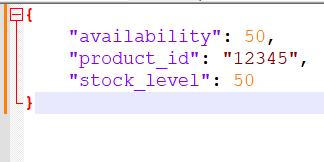
**Sample Response Data:**



**Inventory Management**

* **/api/vendors/{vendor\_id) /inventory/{product\_id}**: GET request to check product availability and stock levels.

**Sample Response Data:**

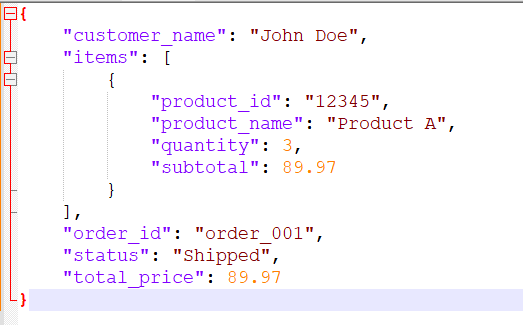


* **/api/vendors/{vendor\_id}/inventory/{product\_id}/adjust**: POST request to allow vendors to adjust product stock levels.

**Order Management**

* **/api/vendors/{vendor\_id}/orders/{order\_id}**: GET request to retrieve specific order details.

**Sample Response Data:**

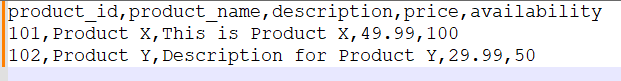


**Bulk Product Import/Export**:

**Import Products**

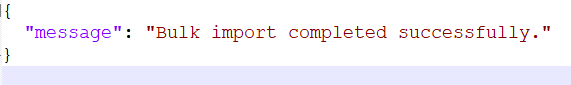
* **/api/vendors/{vendor\_id}/catalogs/{catalog\_id}/import:** POST request to bulk import product data in CSV or Excel format.

**Sample Request Data (CSV Import):**

****

**Sample Request Data (Excel Import):** (not shown as Excel format, but it should follow a similar structure)

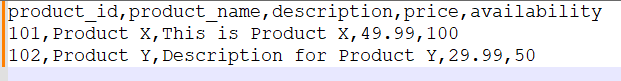
**Sample Response Data:**

****

**Export Products**

* **/api/vendors/{vendor\_id}/catalogs/{catalog\_id}/export:** GET request to export product data in CSV or Excel format.

**Sample Response Data (CSV Export):**

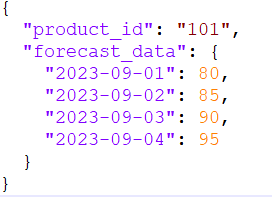
****

**Sample Response Data (Excel Export):** (not shown as Excel format, but it should follow a similar structure)

**Inventory Forecasting**

* **/api/vendors/{vendor\_id}/inventory/{product\_id}/forecast:** GET request to retrieve inventory forecasting data for a specific product.

**Sample Response Data:**

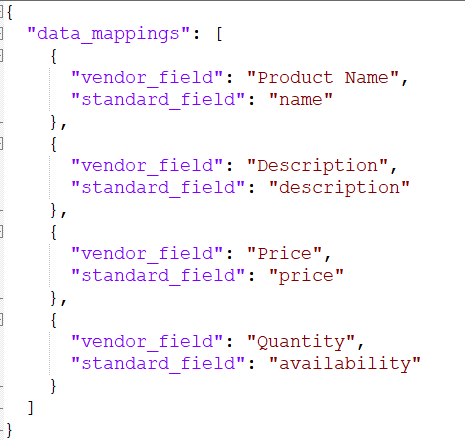
****

**Data Mapping Tool**

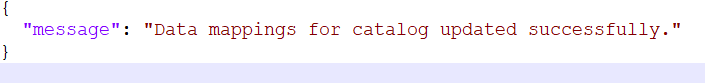
**Create or Update Data Mapping for a Catalog**

* **/api/vendors/{vendor\_id}/catalogs/{catalog\_id}/data-mapping:** POST request allows vendors to create or update data mappings for a specific catalog.

**Sample Request Data**

****

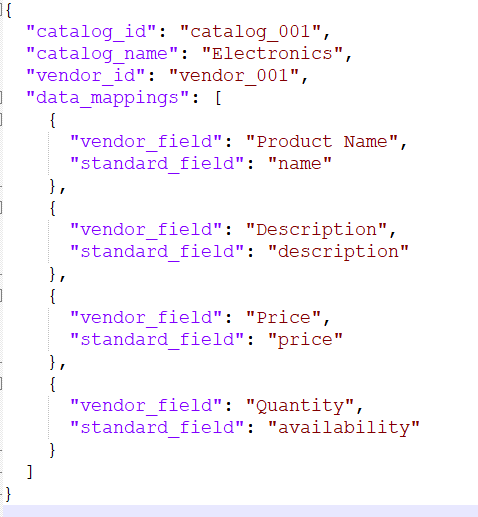
**Sample Response Data**

****

**Get Data Mapping for a Catalog**

* **/api/vendors/{vendor\_id}/catalogs/{catalog\_id}/data-mapping:** GET request retrieves the data mappings for a specific catalog.

**Sample Response Data**:



**Delete Data Mapping for a Catalog**

* **/api/vendors/{vendor\_id}/catalogs/{catalog\_id}/data-mapping:** DELETE Deletes the data mappings for a specific catalog.

**Vendor Notifications**

* **/api/notifications/vendors**: POST request to send notifications to vendors regarding important events or updates.

**Sample Request Data:**

