**EduShelf – Books and Beyond**

**Overview**

**EduShelf: Books and Beyond** is an e-commerce platform designed for books, e-books, stationery, and educational tools. The platform supports **product attributes** specific to educational supplies (e.g., genre, author, edition, language), **subscription-based learning kits** for recurring delivery of tools like science experiment kits, and **e-book downloads** with tiered pricing based on license types.

**Core Functionalities**

1. **Category Management**
   * Organize products into categories like **Books**, **E-books**, **Stationery**, and **Educational Tools**.
   * Subcategories such as **Fiction**, **Non-Fiction**, **STEM Kits**, and **Art Supplies**.
2. **Product Type Management**
   * Define reusable **Product Types** with attributes such as genre, author, edition, and language.
   * Include specific attributes for e-books (e.g., file type and licensing).
3. **Product Management**
   * Create and manage products with attributes such as **edition** (e.g., 2nd Edition), **author**, and **language**.
   * Allow pricing customization for e-books based on license types (e.g., single-user, multi-user).
4. **Product Variant Management**
   * Handle variants for attributes such as language, binding (e.g., paperback, hardcover), or file format (e.g., PDF, ePub).
   * Assign **unique SKUs** to each variant.
5. **Subscription Management**
   * Provide subscription options for recurring deliveries of educational kits (e.g., monthly science kits).
   * Customize delivery frequency (e.g., monthly, quarterly).
6. **E-Book Downloads**
   * Offer download options with tiered pricing for licenses.
   * Manage license-based restrictions for users (e.g., multi-user licenses allow downloads for multiple devices).
7. **Advanced Filtering and Search**
   * Enable filtering by attributes like genre, author, language, and price.
   * Provide sorting options such as most downloaded, highest-rated, and price.

**Entity Relationships**

| **Entity** | **Relationships** |
| --- | --- |
| **Category** | One-to-Many: A category can have multiple subcategories, and products belong to one category. |
| **ProductType** | One-to-Many: A product type defines attributes for multiple products. |
| **Product** | One-to-Many: A product can have multiple variants, and belongs to one category and one product type. |
| **ProductVariant** | Many-to-One: A product variant belongs to one product. |
| **Subscription** | Many-to-Many: Subscriptions can include multiple products, and products can belong to multiple subscriptions. |
| **License** | Many-to-One: Licensing details apply specifically to e-books or software products. |

**Entity Descriptions**

**1. Category**

* Represents a hierarchical structure for browsing educational products.
* **Attributes**:
  + id: Unique identifier.
  + name: Category name (e.g., Books, Stationery).
  + parentCategory: Reference to the parent category for hierarchy.
  + description: Description of the category.

**2. ProductType**

* Defines a template for products with reusable attributes.
* **Attributes**:
  + id: Unique identifier.
  + name: Name of the product type (e.g., Book, E-book).
  + attributes: List of attributes (e.g., Genre, Author, Language, Edition).

**3. Product**

* Represents a specific product linked to a category and product type.
* **Attributes**:
  + id: Unique identifier.
  + name: Product name (e.g., "Introduction to Physics").
  + price: Base price of the product.
  + genre: Genre of the product (e.g., Science, Fiction).
  + author: Author of the product.
  + language: Language of the product (e.g., English, Spanish).
  + categoryId: Reference to the category.
  + productTypeId: Reference to the product type.
  + license: Licensing details for e-books.

**4. ProductVariant**

* Represents specific variations of a product (e.g., language or binding type).
* **Attributes**:
  + id: Unique identifier.
  + sku: Unique SKU for the variant.
  + variantAttributes: Attributes specific to the variant (e.g., Binding: Hardcover, Language: French).
  + stock: Stock quantity for the variant.
  + additionalPrice: Price adjustment for this variant.

**5. Subscription**

* Represents subscription plans for recurring deliveries.
* **Attributes**:
  + id: Unique identifier.
  + name: Subscription name (e.g., Monthly Science Kit).
  + products: List of products included in the subscription.
  + frequency: Delivery frequency (e.g., monthly, quarterly).
  + price: Subscription price.

**6. License**

* Represents licensing options for e-books or digital tools.
* **Attributes**:
  + id: Unique identifier.
  + name: License name (e.g., Single-User, Multi-User).
  + devicesAllowed: Number of devices allowed under the license.
  + priceAdjustment: Additional cost for this license type.

**API Requirements**

**1. Category Management**

**1.1 Create Category**

* **Endpoint**: POST /categories
* **Request Body**:

json

Copy code

{

"name": "Books",

"parentCategoryId": null,

"description": "All types of books including fiction, non-fiction, and educational."

}

* **Response**:

json

Copy code

{

"id": 1,

"name": "Books",

"parentCategoryId": null,

"description": "All types of books including fiction, non-fiction, and educational."

}

**2. Product Type Management**

**2.1 Create Product Type**

* **Endpoint**: POST /product-types
* **Request Body**:

json

Copy code

{

"name": "Book",

"attributes": [

{ "name": "Genre", "type": "Text" },

{ "name": "Author", "type": "Text" },

{ "name": "Edition", "type": "Text" },

{ "name": "Language", "type": "Text" }

]

}

* **Response**:

json

Copy code

{

"id": 1,

"name": "Book",

"attributes": [

{ "name": "Genre", "type": "Text" },

{ "name": "Author", "type": "Text" },

{ "name": "Edition", "type": "Text" },

{ "name": "Language", "type": "Text" }

]

}

**3. Product Management**

**3.1 Create Product**

* **Endpoint**: POST /products
* **Request Body**:

json

Copy code

{

"name": "Introduction to Physics",

"categoryId": 1,

"productTypeId": 1,

"price": 20.00,

"genre": "Science",

"author": "John Doe",

"language": "English",

"license": {

"name": "Single-User",

"devicesAllowed": 1,

"priceAdjustment": 0.00

},

"variants": [

{

"variantAttributes": { "Binding": "Hardcover", "Language": "English" },

"stock": 50,

"additionalPrice": 5,

"sku": "PHYSICS-HARDCOVER-ENG"

},

{

"variantAttributes": { "Binding": "Paperback", "Language": "French" },

"stock": 30,

"additionalPrice": 0,

"sku": "PHYSICS-PAPERBACK-FR"

}

]

}

* **Response**:

json

Copy code

{

"id": 101,

"name": "Introduction to Physics",

"category": "Books",

"price": 20.00,

"genre": "Science",

"author": "John Doe",

"language": "English",

"license": {

"name": "Single-User",

"devicesAllowed": 1,

"priceAdjustment": 0.00

},

"variants": [

{

"id": 201,

"variantAttributes": { "Binding": "Hardcover", "Language": "English" },

"stock": 50,

"additionalPrice": 5,

"sku": "PHYSICS-HARDCOVER-ENG"

},

{

"id": 202,

"variantAttributes": { "Binding": "Paperback", "Language": "French" },

"stock": 30,

"additionalPrice": 0,

"sku": "PHYSICS-PAPERBACK-FR"

}

]

}

**Entities and Relationships**

**1. ProductType**

Represents reusable templates for product attributes.

java

Copy code

@Entity

public class ProductType {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String name; // e.g., Book, E-Book, Educational Kit

@OneToMany(cascade = CascadeType.ALL, orphanRemoval = true)

@JoinColumn(name = "productType\_id") // Foreign key for attributes

private List<AttributeDefinition> attributes = new ArrayList<>();

}

**2. AttributeDefinition**

Defines attribute names, types, and constraints for ProductType.

java

Copy code

@Entity

public class AttributeDefinition {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String name; // e.g., Genre, Author, Language

@Enumerated(EnumType.STRING)

private AttributeType type; // ENUM, TEXT, NUMBER, etc.

@ElementCollection

private List<String> allowedValues; // For ENUM attributes (e.g., ["PDF", "EPUB", "Paperback"])

}

**3. Product**

Represents a specific product linked to a ProductType and Category.

@Entity

public class Product {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String name;

private String description;

private BigDecimal basePrice;

@ManyToOne

@JoinColumn(name = "productType\_id", nullable = false)

private ProductType productType;

@OneToMany(mappedBy = "product", cascade = CascadeType.ALL, orphanRemoval = true)

private List<ProductVariant> variants = new ArrayList<>();

@ManyToMany

@JoinTable(

name = "product\_category",

joinColumns = @JoinColumn(name = "product\_id"),

inverseJoinColumns = @JoinColumn(name = "category\_id")

)

private List<Category> categories = new ArrayList<>();

}

**4. ProductVariant**

Represents specific variations of a product.

@Entity

public class ProductVariant {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String sku; // Unique identifier

private BigDecimal price; // Price for this variant

@ElementCollection

private Map<String, String> variantAttributes; // E.g., {"Binding": "Hardcover", "Language": "English"}

private Integer stock; // Available stock

@ManyToOne

@JoinColumn(name = "product\_id", nullable = false)

private Product product;

}

**5. License**

Defines licensing options for e-books or digital products.

@Entity

public class License {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String type; // Single-user, Multi-user

private Integer allowedDevices; // Number of devices allowed

private BigDecimal priceAdjustment; // Price adjustment for the license

}

**6. Subscription**

Represents recurring delivery plans for products.

@Entity

public class Subscription {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String name; // E.g., Monthly Science Kit

private BigDecimal price;

private String frequency; // Monthly, Quarterly

@ManyToMany

private List<Product> products = new ArrayList<>();

}