# V-Mart-Ecom

⇨ **Function Based API View Project**

⇨ **Aim of this Ecom project is to selling Verities of Items In categories Electronic, Clothes, Fashion & Beauty, Packaged Foods, Home & kitchenware, Footwear, Stationary, Children were & Gifts, Etc..**

**Roles and Permission**

**1. Seller**

**Role: Marketplace vendor who lists and sells products.**
**Permissions:**

- **Product Management**

  o **Create, update, and delete their own products**

  o **View their product inventory and sales analytics**

- **Order Management**

  o **View orders containing their products**

  o **Update order status (e.g., mark as "Shipped" or "Delivered")**

  o **Process/handle refund requests**

- **Profile Management**

  o **Register as a seller (after admin verification)**

  o **Update their profile (contact, address, etc.)**

  o **Cannot delete other users or access admin features**

**Restrictions:**

- **Cannot access other sellers' products or orders**

- **Cannot modify (categories, subcategories, users, etc.)**

## 2. Buyer

**Role: Customer who purchases products.**
**Permissions:**

- **Shopping Features**

    o **Browse/search products**

    o **Add items to cart/wishlist**

    o **Place orders and make payments**

- **Order Management**

    o **View their own order history**

    o **Cancel orders (if allowed)**

    o **Request refunds/returns**

- **Review System**

    o **Rate and review purchased products**

    o **Edit/delete their own reviews**

- **Profile Management**

    o **Register and update their account details**

    o **Cannot delete other users or access seller/admin features**

**Key Restrictions:**

- **Cannot list products for sale**

- **Cannot view other buyers' orders or data**

## 3. Admin (Superuser/Staff)

**Role: Platform administrator with full control.**
**Permissions:**

- **User Management**

- View, verify, or ban sellers/buyers
- Delete user accounts

- **Catalog Management**

  - Create/update/delete product categories and subcategories

  - Edit/remove any product (even from other sellers)

- **Order & Payment Oversight**

  - View all orders across the platform

  - Force-cancel orders or override refunds

- **System Configuration**

  - Modify global settings (e.g., shipping rules, fees)

  - Access analytics/reports

**Key Restrictions:**

- Cannot purchase products (unless also registered as a buyer)

**List of Functions**

**Authentication & User Management**

1. login() - User authentication (JWT tokens)

2. logout_view() - Token invalidation

3. forgot_password() - Password reset

4. register_seller() - Seller registration

5. update_seller() - Seller profile update

6. seller_delete() - Seller account deletion

7. seller_profile_with_products_orders() - Seller dashboard

8. register_buyer() - Buyer registration

9. update_buyer() - Buyer profile update

10. buyer_delete() - Buyer account deletion

11. buyer_profile_with_orders() - Buyer dashboard

12. admin_all_buyers() - Admin view of all buyers

13. admin_all_sellers() - Admin view of all sellers

## Category Management

14. category_get() - Get categories

15. category_create() - Create category (admin)

16. category_update() - Update category (admin)

17. category_delete() - Delete category (admin)

## Subcategory Management

18. subcategory_get() - Get subcategories

19. subcategory_create() - Create subcategory (admin)

20. subcategory_update() - Update subcategory (admin)

21. subcategory_delete() - Delete subcategory (admin)

## Product Operations

22. product_get() - Get products

23. product_search() - Advanced product search

24. product_create() - Create product (seller)

25. product_update() - Update product (seller)

26. product_delete() - Delete product (seller)

## Cart Operations

27. cart_get() - View cart contents

28. cart_create() - Add to cart

29. cart_items_update() - Update cart item quantity

30. **cart_delete() - Empty cart**

## Order Processing

31. **create_order() - Create order from cart**

32. **order_list() - Buyer order history**

33. **seller_order_list() - Seller order view**

34. **update_order_status() - Update order status (seller)**

35. **cancel_order_and_refund() - Cancel order(buyer)**

36. **update_refund_status() - Process refunds (seller)**

## Review System

37. **create_review() - Create product review**

38. **update_review() - Update review**

39. **delete_review() - Delete review**

## Wishlist Management

40. **wishlist_create() - Add to wishlist**

41. **wishlist_remove() - Remove from wishlist**

## Utility Functions

42. **generate_order_id() - Generates unique order IDs**

43. **valid_email() - Email validation helper**

44. **get_seller() - Seller retrieval helper**


## Models

⇨ **This fields will automatically added in table…**

⇨ **'created_at', 'updated_at', 'created_by', 'created_by'**

## User

- **Extends Django's AbstractUser**

- **Used for authentication**

**Seller**

- **OneToOne relation with User**
- **Fields: profile_picture, mobile_no, address, is_verified,**

**Buyer**

- **OneToOne relation with User**
- **Fields: profile_picture, mobile_no, address**

**Category**

- **Fields: category_name, category_detail**

**SubCategory**

- **ForeignKey to Category**
- **Fields: subcategory_name, subcategory_detail**

**Product**

- **ForeignKey to Seller, Category, SubCategory**
- **Fields: name, description, price, sale_price, quantity, brand, etc.**

**Cart**

- **OneToOne relation with Buyer**
- **Contains CartItems**

**Order**

- **ForeignKey to Buyer, Payment**
- **Status choices: PENDING, PROCESSING, SHIPPED, DELIVERED, CANCELLED, REFUNDED**

**ProductReview**

- **ForeignKey to Product, Buyer**
- **Fields: rating (1-5), comment**

**Wishlist**

- **ForeignKey to Buyer, Product**

- **Tracks buyer's wishlisted products**

**Login**

- **Endpoint: POST /api/login/**

- **Description: Authenticates a user and returns JWT tokens**

- **Request Body:**

```
{
  "username": "string",
  "password": "string"
}
```

**Logout**

- **Endpoint: POST /api/logout/**

- **Description: Bloklist the token**

- **Request Body:**

```
{
  "refresh": "string (refresh token)"
}
```

**Forgot Password**

- **Endpoint: POST /api/forgot_password/**

- **Description: Resets user password**

- **Request Body:**

```
{
  "email": "string",
  "new_password": "string"
}
```

# User Management

**Seller**

**Register Seller**

- **Endpoint: POST /api/seller_register/**

- **Description: Creates a new seller account**

- **Required Fields: username, email, password, mobile_no, address**

- **Request Body:**

```
{
  "username": "string",
  "email": "string",
  "password": "string",
  "mobile_no": "string (10 digits)",
  "address": "string",
  "profile_picture": "file (optional)"
}
```

**Update Seller**

- **Endpoint: PUT /api/update_seller/**

- **Description: Updates seller profile**

- **Authentication: Required**

- **Request Body:**

```
{
  "seller_id": "int",
  "username": "string (optional)",
  "email": "string (optional)",
  "password": "string (optional)",
  "mobile_no": "string (optional)",
  "address": "string (optional)",
  "profile_picture": "file (optional)"
}
```

**View Seller Profile**

- **Endpoint: GET /api/seller_profile_with_products_orders/**

- **Description: Returns seller profile with products and orders**

- **Authentication: Required (seller only)**

## Buyer

## Register Buyer

- **Endpoint: POST /api/buyer_register/**

- **Description: Creates a new buyer account**

- **Required Fields: username, email, password, mobile_no, address**

- **Request Body:**

```
{
  "username": "string",
  "email": "string",
  "password": "string",
  "mobile_no": "string (10 digits)",
  "address": "string",
  "profile_picture": "file (optional)"
}
```

## Update Buyer

- **Endpoint: PUT /api/update_buyer/**

- **Description: Updates buyer profile**

- **Authentication: Required**

- **Request Body:**

```
{
  "buyer_id": "int",
  "username": "string (optional)",
  "email": "string (optional)",
  "password": "string (optional)",
  "mobile_no": "string (optional)",
  "address": "string (optional)",
  "profile_picture": "file (optional)"
}
```

## View Buyer Profile

- **Endpoint: GET /api/buyer_profile_with_orders/**

- **Description: Returns buyer profile with order history**

- **Authentication: Required (buyer only)**

## Category Management

## Get Categories

- **Endpoint: GET /api/category_get/**

- **Description: Returns all categories or single category if ID provided**

- **Parameters:**

  - pk (optional): Category ID

## Create Category

- **Endpoint: POST /api/category_create/**

- **Description: Creates a new category (admin only)**

- **Authentication: Admin required**

- **Request Body:**

```
{
  "category_name": "string",
  "category_detail": "string (optional)"
}
```

## Update Category

- **Endpoint: PUT /api/category_update/**

- **Description: Updates category details (admin only)**

- **Authentication: Admin required**

- **Request Body:**

```
category_id": "int",
  "category_name": "string",
  "category_detail": "string (optional)"
}
```

**Delete Category**

- **Endpoint: DELETE /api/category_delete/**

- **Description: Deletes a category (admin only)**

- **Authentication: Admin required**

- **Parameters:**

    - **pk: Category ID**

**Product Management**

**Get Products**

- **Endpoint: GET /api/product_get/**

- **Description: Returns all products or single product if ID provided**

- **Parameters:**

    - **product_id (optional): Product ID**

**Search Products**

- **Endpoint: POST /api/product_search/**

- **Description: Advanced product search with filters**

- **Request Body:**

```
{
    "name": "string (optional)",
    "category": "string (optional)",
    "sub_category": "string (optional)",
    "price": "decimal (optional)",
    "brand": "string (optional)",
    "color": "string (optional)",
    "size": "string (optional)",
    "sort_by": "price_low|price_high|newest (optional)"
}
```

**Create Product**

- **Endpoint: POST /api/product_create/**

- **Description: Creates a new product (seller only)**

- **Authentication: Seller required**

- **Request Body:**

```json
{
  "name": "string",
  "description": "string",
  "price": "decimal",
  "sale_price": "decimal (optional)",
  "quantity": "int",
  "category": "int (category ID)",
  "sub_category": "int (subcategory ID)",
  "brand": "string",
  "tags": "string (optional)",
  "size": "string (optional)",
  "color": "string (optional)",
  "fabric": "string (optional)"
}
```

## Update Product

- **Endpoint: PUT /api/product_update/**

- **Description: Updates product details (seller only)**

- **Authentication: Seller required**

- **Request Body:**

```json
{
  "product_id": "int",
  "name": "string (optional)",
  "description": "string (optional)",
  "price": "decimal (optional)",
  "sale_price": "decimal (optional)",
  "quantity": "int (optional)",
  "category": "int (optional)",
  "sub_category": "int (optional)",
  "brand": "string (optional)",
  "tags": "string (optional)",
  "size": "string (optional)",
  "color": "string (optional)",
  "fabric": "string (optional)"
}
```

## Delete Product

- **Endpoint: DELETE /api/product_delete/**

- **Description: Deletes a product (seller only)**

- **Authentication: Seller required**

- **Request Body:**

```json
{
  "product_id": "int"
}
```

## Cart Operations

## Get Cart

- **Endpoint: GET /api/cart_get/**

- **Description: Returns cart contents**

- **Authentication: Buyer required**

- **Request Body:**

```json
{
  "cart_id": "int"
}
```

## Add to Cart

- **Endpoint: POST /api/cart_create/**

- **Description: Create and Adds product to cart**

- **Authentication: Buyer required**

- **Request Body:**

```json
{
  "buyer_id": "int",
  "product_id": "int",
  "quantity": "int",
  "color": "string (optional)",
  "size": "string (optional)"
}
```

## Update Cart Item

- **Endpoint: PUT /api/cart_items_update/**

- **Description: Updates cart item quantity**

- **Authentication: Buyer required**

- **Request Body:**

```
{
  "cart_item_id": "int",
  "quantity": "int"
}
```

## Delete Cart

- **Endpoint: DELETE /api/cart_delete/**

- **Description: Deletes entire cart**

- **Authentication: Buyer required**

- **Request Body:**

```
{
  "cart_id": "int"
}
```

## Order Processing

## Create Order

- **Endpoint: POST /api/create_order/**

- **Description: Creates order from cart**

- **Authentication: Buyer required**

- **Request Body:**

```
{
  "payment_method": "COD|CREDIT_CARD|DEBIT_CARD|NET_BANKING|UPI",
  "transaction_id": "string",
  "delivery_address": "string",
  "delivery_contact": "string (10 digits)"}
```

## View Orders

- **Endpoint: GET /api/order_list/**

- **Description: Returns buyer's order history**

- **Authentication: Buyer required**

- **Parameters:**

    - status: Filter by status (optional)

## Update Order Status

- **Endpoint: PUT /api/update_order_status/**

- **Description: Updates order status (seller only)**

- **Authentication: Seller required**

- **Request Body:**

```
{
  "order_number": "string",
  "status": "PENDING|PROCESSING|SHIPPED|DELIVERED|CANCELLED|REFUNDED",
  "tracking_number": "string (optional)",
  "shipping_company": "string (optional)",
  "notes": "string (optional)"
}
```

## Cancel Order

- **Endpoint: PUT /api/cancel_order_and_refund/**

- **Description: Cancels order and initiates refund**

- **Authentication: Buyer required**

- **Request Body:**

```
{
  "order_number": "string",
  "cancel_reason": "string"
}
```

## Reviews

## Create Review

- **Endpoint: POST /api/create_review/**

- **Description: Creates product review**

- **Authentication: Buyer required**

- **Request Body:**

```
{
    "product_id": "int",
    "rating": "int (1-5)",
    "comment": "string (optional)"
}
```

## Update Review

- **Endpoint: PUT /api/update_review/**

- **Description: Updates product review**

- **Authentication: Buyer required**

- **Request Body:**

```
{
    "review_id": "int",
    "rating": "int (1-5, optional)",
    "comment": "string (optional)"
}
```

## Delete Review

- **Endpoint: DELETE /api/delete_review/**

- **Description: Deletes product review**

- **Authentication: Buyer required**

- **Request Body:**

```
{
    "product_review_id": "int"
}
```

## Wishlist

## Add to Wishlist

- **Endpoint: POST /api/wishlist_create/**

- **Description: Adds product to wishlist**

- **Authentication: Buyer required**

- **Request Body:**

```
{
  "product_id": "int"
}
```

## Remove from Wishlist

- **Endpoint: DELETE /api/wishlist_remove/**

- **Description: Removes product from wishlist**

- **Authentication: Buyer required**

- **Request Body:**

```
{
  "wishlist_id": "int"
}
```