We are following Biswanath Giri's example. [1]
About the tables:

- The **Users** document should contain the user's name, email, role, sizes of upper garments, lower garments and other garments, his shipping addresses and date_created_at. The **Users** document will be linked to the **CartItems** entity which will contain cart_item_id**,** cart_item_image, cart_item_price and cart_item_quantity. The **Users** document should be linked to the **Orders** document. The **Users** document should be linked to the **ShippingAddresses** entity which will have the shipping_address_id and the …

- The **Orders** document should contain the order's id, the user's id, its status, shipping_address and total_price, date_created_at and date_of_arrival. It should be linked to the **OrderItems** entity which will contain the order_item_id, product_id, image_uri, user_id and quantity.

- The **Products** document should contain information on each product such as its name, images, color, garment_material, fit, specifications, supply_type and date_created_at. It should be linked to the **ProductItems** entity that contains the sku, (inseam_length, OR upper_size_number and upper_size_letter OR other_size_number and other_size_letter), current_stock, total_stock and quantity_sold. The **Products** document should also be linked to the **ProductImages** entity that contains the product_image_ids and URIs of the respective product's images. The **Products** document should also be linked to the **ProductReviews** entity that contains the product_review_id, product_review_comment, date_created_at and product_review_rating of all the reviews that exist for that product. The **Products** document should also be linked to the **CartItems** document which will contain cart_item_id**,** cart_item_image, cart_item_price and cart_item_quantity.

- The **Categories** and **Subcategories** entities aren't going to have their own documents as these are single-valued attributes. So, even though they have a 1:N relationship with products (A category / subcategory can be had by N products), they aren't getting their own documents. A color can also be shared by many products. This doesn't mean that it gets its own document.

- The **UpperSizes**, **LowerSizes** and **OtherSizes** entities won't have their own documents since they are easier to store through embedding since they don't have an N relationship with any entity.

Entity	Attribute	Relationship
Users	user_id	1:1, A user has 1 user_id.
	date_created_at	1:1, A user has one date_created_at.
	email	1:1, A user has 1 email.
	password	1:1, A user has 1 password.
	phone_number	1:1, A user has 1 phone_number.
	first_name	1:1. A user has one firstname.
	last_name	1:1, A user has has one lastname
	user_role	1:1, A user has one user_role.
	upper_size_number	1:1, A user has one upper_size_number.
	upper_size_letter	1:1, A user has one upper_size_letter.
	lower_size_number	1:1, A user has one lower_size_number.
	lower_size_letter	1:1, A user has one lower_size_letter.
	others_size_number	1:1, A user has 1 others_size_number.
	others_size_letter	1:1, A user has one others_size_letter.
	email_comms_type	1:1, A user has one email_comms_type.
	sms_comms	1:1, A user has one sms_comms.
	SHIPPING_ADDRESS	**1: 0 or N**, A user can have 0 or N shipping addresses. An address belongs to 1 user.
	ORDER	**1: 0 or N**, A user can have 0 or N orders. An order belongs to 1 user.

	CARTITEM	**1: 0 or N**, A user can have 0 or N cart items. A cart item belongs to 1 user.
Orders	order_id	1:1. An order has only 1 order_id.
	ORDER_ITEM	**1:1 or N**, An order can have 1 or N order_items.
	order_status	1:1, An order can have one order_status.
	total_price	1:1, An order has one total_price.
	date_created_at	1:1, An order is created at a unique date.
	date_of_arrival	1:1, An order arrives at a particular date.
	shipping_address	1:1, An order can have 1 shipping address.
OrderItems	order_item_id	1:1, An order_item has an id.
	order_item_price	1:1, An order_item has 1 price.
	order_item_quantity	1:1, An order item has 1 quantity.
	order_item_image_uri	1:1, An order item has 1 image.
	PRODUCT	**1:1**, An order item **IS** 1 product. However, **1 order can have N products**. Since we can't put product_id in the Orders table (there, it can't form a unique row since order items aren't there), so we put it here.
Products	product_id	1:1, A product has 1 product_id.
	product_name	1:1, A product has 1 product_name.
	product_type	1:1, A product has 1

		product_type.
	product_color	1:1, A product has 1 product_color.
	product_description	1:1, A product has 1 product_description.
	product_price	1:1, A product has 1 product_price.
	REVIEW	**1:N**, A product has N reviews. Each review belongs to 1 product.
	product_status	*1:1, A product has one product_status.*
	IMAGE	**1:N**, A product has N images. An image belongs to 1 product.
	product_category	1:1, A product has 1 product_category.
	product_subcategory	1:1, A product has 1 product_subcategory.
	product_fit	1:1, A product has 1 product_fit.
	product_specifications	1:1, A product has only 1 set of product_specifications.
	product_garment_weight	1:1, A product has only 1 product_garment_weight.
	product_material	1:1, A product has only 1 product_material.
	product_supply_type	1:1, A product has only 1 nsupply_type.
	PRODUCT_ITEMS	**1: 0 or N**. Each product will have 0 or more product items.
ProductItems	sku	1:1, Each product item has an sku.
	inseam_length	1: 0 or 1. Each product item has 0 or 1 inseam length.
	upper_size_number	1:0 or 1. Each product item has 0 or 1 upper_size_number.

	upper_size_letter	1:0 or 1. Each product_item has 0 or 1 upper_size_letter.
	lower_size_number	1:) or 1, Each product_item has 0 or 1 lower_size_number.
	other_size_letter	1:0 or 1. Each product_item has 0 or 1 other_size_letter.
	total_stock	1:1, Each product item has a total_stock.
	current_stock	1:1, Each product item has a current_stock.
	quantity_sold	1:1, Each product item has a quantity_sold.
ProductImages	image_id	1:1, A product image has only 1 image_id.
	image_uri	1:1, A product_image has only 1 image_uri.
	main_image	1:1, A product_image has 1 main_image.
CartItems	cart_item_id	1:1, A cart_item has 1 cart_item_id.
	cart_item_image	1:1, A cart_item has 1 cart_item_image.
	cart_item_price	1:1, A cart_item has 1 cart_item_price.
	cart_item_quantity	1:1, A cart_item has 1 cart_item_quantity.
ShippingAddress	street_number	1:0 or 1, An address has 0 or 1 street number
	building_name	1:0 or 1, An address has 0 or 1 building_name.
	Street_number_suffix [A, B, etc]	1:0 or 1, An address has 0 or 1 street_number_suffix.
	Street_name	1:0 or 1, An address has 0 or 1 street_name_suffix.
	Street_type [Court, Avenue, Drive, etc.]	1:0 or 1, An address has 0 or 1 street_direction.
	Street_direction	1:0 or 1

	[N,W,E,S,SE,NE,NW,SW]	
	Address_type_id [Room Number, Office Number, etc.]	1:0 or 1, An address has 0 or 1 address_type.
	City / Town	1: 0 or 1
	Postal Area (Eg - ZIP, Postcode, PIN, P.O. Box, etc.]	1:0 or 1
	Country	1:0 or 1
Reviews	review_id	1:1, Each review has a review_id.
	reported_fit	1:1, Each review has a reported_fit.
	review_comment	1:1, Each review has a review_comment.
	review_rating	1:1, Each review has a review_rating.

REFERENCES

[1]
https://bgiri-gcloud.medium.com/designing-the-database-schema-for-a-new-e-commerce-platform-and-considering-factors-like-ec28d4fb81db

NOTES

- We can add a supplier tracking system in the future.
- We can make the inventory process more detailed such that it can track inflow of fresh stocks of products to the inventory.
- Each product shares the same price with all its product items.