Order of Operations

Table	Insert	Update	Delete
USERS	-	-	Delete all REVIEWS of that USER. Delete all ORDERS of that USER. ShippingAddresses are automatically deleted. CartItems are automatically deleted.
ShippingAddresses	Check if the user exists (**user_id**). A ShippingAddress is inserted into USER.	Check if the user exists (**user_id**). A ShippingAddress is updated in USER.	Check if the user exists (**user_id**). A ShippingAddress is deleted from USER.
CartItems	Check respective PRODUCT and ProductItem before inserting a CartItem [**product_id, sku**]. Set the **cart_item_price** based on **product_price** in the respective PRODUCT. Set the **cart_item_quantity** based on **current_stock** in the respective ProductItem**.** *Set the **cart_item_image** based on the **main_image** in ProductImages in the respective PRODUCT.* A CartItem is inserted	Check respective PRODUCT and ProductItem before inserting a CartItem [**product_id, sku**]. Update the **cart_item_price** based on **product_price** in the respective PRODUCT. Update the **cart_item_quantity** based on **current_stock** in the respective ProductItem**.** *When updating the **cart_item_image**, do so by checking the **main_image** in ProductImages in the respective PRODUCT.* A CartItem is updated	A CartItem is deleted from USER.

	into USER.	in USER.	
ORDERS	Check the respective USER before inserting an ORDER. [**user_id**]. Check whether the respective USER has a non-empty **CartItems** array. The inserted ORDER must have at least 1 OrderItem.	An order can be updated only if its **order_status** is “Pre-Dispatch”. The **order_status**, **date_of_arrival** can only be updated by **admin** users. The **shipping_address** in order can be updated only if the **order_status** is **Pre-Dispatch**.	When deleting an ORDER, delete all of its OrderItems. An ORDER can be deleted by the user only if its status is “Pre-Dispatch”.
OrderItems	Check that the **order_status** is **Pre-Dispatch**. Check the respective CartItem before inserting an OrderItem. [**product_id, sku, product_price**, **current_stock**]. Insert OrderItem in ORDER. Delete the respective CartItem. Update the **total_price** in the respective ORDER. Then, update the **total_stock**, **quantity_sold** and **current_stock** fields	Check the respective ProductItem, PRODUCT before updating the **product_id**, **sku** or **order_item_quantity** of an OrderItem. [**product_id, sku, current_stock**]. Check that the **order_status** is **Pre-Dispatch**. Update OrderItem in ORDER. Update the **total_price** in the respective ORDER. Then, update the **total_stock**, **quantity_sold** and **current_stock** fields in the respective	Check that the **order_status** is **Pre-Dispatch**. Delete the OrderItem from ORDER. Update the **total_price** in the respective ORDER. Then, update the **total_stock**, **quantity_sold** and **current_stock** fields in the respective ProductItem.

	in the respective ProductItem.	ProductItem.	
NOTE: Order items can be added or deleted by all types of users, provided ***they are authorized*** and that the order is "**Pre-Dispatch"**. The same is true of cancelling orders. **NOTE: order_item_price** and **order_image_uri** of OrderItem can't be updated there.			
PRODUCTS	When inserting a PRODUCT, insert at least 1 ProductItem and 1 ProductImage.	When updating a PRODUCT [**product_id, product_price**], update the corresponding CartItems [**product_id**, **cart_item_price**] *If updating a PRODUCT's **product_id,** then we need to update the same across corresponding OrderItems, CartItems and REVIEWS.*	When deleting a PRODUCT: Delete all ProductImages of that PRODUCT. Delete all REVIEWS of that PRODUCT. Delete all CartItems of that PRODUCT. Delete all ProductItems of that PRODUCT.
ProductImages	Insert ProductImage in PRODUCT.	Update ProductImage in PRODUCT. Update **cart_item_image_uri** in respective CartItem, if the ProductImage updated is the main image. Update **order_item_image_ uri** in OrderItem if the ProductImage updated is the main image.	Delete ProductImage from PRODUCT. Delete **cart_item_image** in respective CartItem, if the ProductImage deleted is the main image. Delete **order_item_image_ uri** in OrderItem if the ProductImage deleted is the main image.

ProductItems	Insert ProductItem in PRODUCT.	Update ProductItem in PRODUCT *Update corresponding OrderItem in ORDER (Optional).* Update corresponding CartItem in USER. (**cart_item_quantity**) if the **current_stock** falls below the **cart_item_quantity**. Do this globally.	Delete CartItem in USER. *Delete OrderItem in ORDER.* Delete ProductItem in PRODUCT.
REVIEWS	When inserting a REVIEW, check whether the corresponding PRODUCT and ProductItem exist. [**product_id, sku**] REVIEW is inserted in USER.	When updating the **product_id** or **sku** of a REVIEW, check whether the corresponding PRODUCT and ProductItem exist. [**product_id, sku**]. REVIEW is updated in USER.	REVIEW is deleted from USER.

NOTE: ProductImage isn't linked to **OrderItem** as it may continue to exist even after the corresponding ProductItem is deleted.

NOTE: ProductImage isn't linked to **CartItem** since in the future, CartItem will be in Redis.

NOTE: This chart doesn't show what would happen if we tried to update primary keys.