Order Management API – Business Logic

# Customers

* E-mail must be unique.
  + If the E-mail exists in the database, then the user is not created.
  + The response should give a generic error message for security reasons.
* Phone number must be unique
  + If the Phone number exists in the database, then the user is not created.
  + The response should give a generic error message for security reasons.
  + Only supports phone numbers in Denmark and should therefore be 8 digits.
* Required fields
  + FirstName, LastName, Address, Mail
* Customer Status
  + NoAccount
    - The user explicitly said they do NOT wish to create an account.
    - The user is still created, but with fillers on required fields.
  + Customer
    - The user signed up, but did not place an order yet
  + PurchasedOnce
    - The user is signed up and purchased an order once.
    - Eligible for discount code on next order
  + PurchasedMultiple
    - The user is signed up and purchased multiple orders.
    - Eligible for special discounts based purchased products.
  + Deactivated
    - The user has deactivated their account.
    - Not able to sign in as a user but is still registered in the system.

# Orders

* Calculate subtotal
  + Each orderline on the order should include a subtotal (quantity \* product.price)
* Calculate total price
  + The total price of the order, including shipping and handling fees
  + The shipping and handling fee should be included above, separately.
* Validate ShippedTo has a value
  + The ShippedTo is the customer address, unless something else is specified
* Validate Order Information on Create
  + If the customer status is NoAccount, this part is skipped
  + Is the order null?
  + Does the order have orderlines?
* Validate Updating an Order
  + If the order status is received or processing the order can be updated, else return
  + If the order has no id, return
* Validate Order is Shipped
  + If the order status is Shipped and ShippedOn has value, else return
* The DateTime properties
  + **PlacedOn:** When the order is created by the customer, the date is set
  + **ProcessedOn:** When the order is going to be processed by the admin, the date is set
  + **ShippedOn:** When the order is being shipped by the admin, the date is set
* Order Status
  + Received
    - The order is received and the PlacedOn date is set
    - The order is not yet being processed and can still be cancelled
  + Processing
    - The order is now being processed and the ProcessedOn date is set
    - The order is not yet shipped and can still be cancelled
      * If cancelled, the shipping and handling fee is not returned
  + Shipped
    - The order is now being shipped and the ShippedOn date is set
    - The order has left the storage and can NOT be cancelled
    - The order can still be returned
  + Returned
    - The order has been sent and the customer returned the order.
  + Cancelled
    - The order is still in the warehouse and the customer cancelled the order.

# Products

* Required Fields
  + Name, Description, stock, price
* Product Status
  + In Stock
    - The product is currently in stock and purchasable
  + Out of Stock
    - The product is sold out and has not been ordered to stock by admin
  + In Order
    - The product is out of stock, but has been ordered to stock by admin
  + Expired
    - The product is expired from the product catalogue
    - The product is sold until out of stock