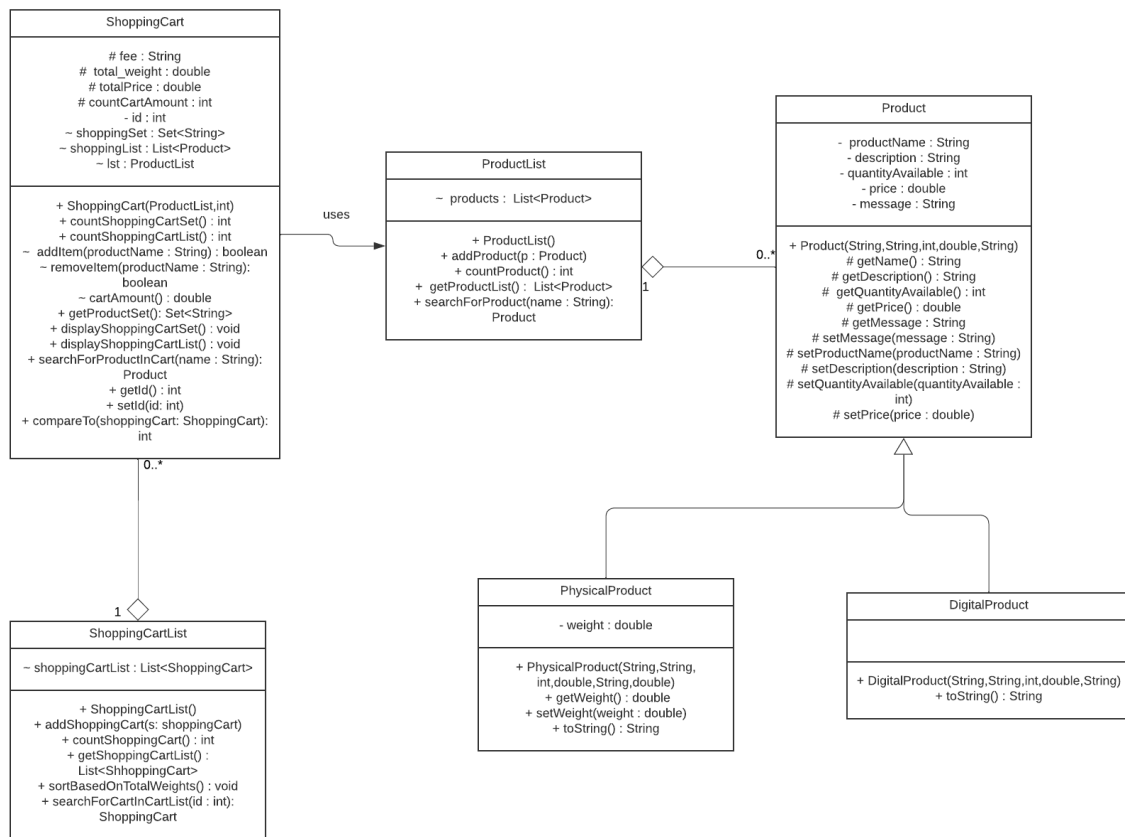# Technical report

## Student: Doan Duy Bach – S3926953

1. Description of the different OOP principles utilized in my project.
- In my solution, I used encapsulation to shield the data stored in the class from system-wide access by keeping numerous fields private. For example, the Product class is completely encapsulated since it contains five private attributes, and each field has its own setter and getter methods. This keeps the detailed information related to each product hidden from view outside of the product itself. Therefore, the classes outside have to create new Product objects and use setter and getter methods to view and change those fields. In the project code, I apply inheritance by extending the Product class with two child classes: PhysicalProduct and DigitalProduct. That implements the Don't Repeat Yourself principle and in the PhysicalProduct class, it helps define an extra field called weight which represents the weight of the physical product. Moreover, the super keyword is used in PhysicalProduct and DigitalProduct to explicitly access different attributes in the Product class. Besides, in those child classes, the toString() method is overridden to display different String representations.

2. The UML class diagram that describe my solution:

**ShoppingCart**

```
# fee : String
# total_weight : double
# totalPrice : double
# countCartAmount : int
- id : int
~ shoppingSet : Set<String>
~ shoppingList : List<Product>
~ lst : ProductList

+ ShoppingCart(ProductList,int)
+ countShoppingCartSet() : int
+ countShoppingCartList() : int
~ addItem(productName : String) : boolean
~ removeItem(productName : String):
  boolean
~ cartAmount() : double
+ getProductSet(): Set<String>
+ displayShoppingCartSet() : void
+ displayShoppingCartList() : void
+ searchForProductInCart(name : String):
  Product
+ getId() : int
+ setId(id: int)
+ compareTo(shoppingCart: ShoppingCart):
  int
```

**ProductList**

```
~ products : List<Product>

+ ProductList()
+ addProduct(p : Product)
+ countProduct() : int
+ getProductList() : List<Product>
+ searchForProduct(name : String):
  Product
```

**Product**

```
- productName : String
- description : String
- quantityAvailable : int
- price : double
- message : String

+ Product(String,String,int,double,String)
# getName() : String
# getDescription() : String
# getQuantityAvailable() : int
# getPrice() : double
# getMessage() : String
# setMessage(message : String)
# setProductName(productName : String)
# setDescription(description : String)
# setQuantityAvailable(quantityAvailable :
  int)
# setPrice(price : double)
```

**ShoppingCartList**

```
~ shoppingCartList : List<ShoppingCart>

+ ShoppingCartList()
+ addShoppingCart(s: shoppingCart)
+ countShoppingCart() : int
+ getShoppingCartList() :
  List<ShhoppingCart>
+ sortBasedOnTotalWeights() : void
+ searchForCartInCartList(id : int):
  ShoppingCart
```

**PhysicalProduct**

```
- weight : double

+ PhysicalProduct(String,String,
  int,double,String,double)
+ getWeight() : double
+ setWeight(weight : double)
+ toString() : String
```

**DigitalProduct**

```
+ DigitalProduct(String,String,int,double,String)
+ toString() : String
```

## 3. Working screenshots of functions of my solution code:
- Three Screenshot of creating two physical products and one digital product:



```
admin@MacBook-Pro IndividualProject %  /usr/bin/env /Library/Java/JavaVirtualMachines/jdk-19.jdk/Contents/Home/bin/jav
a -XX:+ShowCodeDetailsInExceptionMessages @/var/folders/k4/q7mk2rf574z9t358r0_1ngpm0000gn/T/cp_5ncih2yxvsdtjxfld91qa91
ph.argfile vn.edu.rmit.Main
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 0

Enter the type of the product:(physical or digital):
physical
Enter the name of the product:
p
Enter the description of the product:
physical product
Enter the message of the product:
good morning
Enter the available quantity of the product:
10
Enter the price of the product:
10
Enter the weight of the product:
5
```

```
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 0

Enter the type of the product:(physical or digital):
digital
Enter the name of the product:
d
Enter the description of the product:
digital product
Enter the message of the product:

Enter the available quantity of the product:
13
Enter the price of the product:
14
```

```
Enter the type of the product:(physical or digital):
physical
Enter the name of the product:
p1
Enter the description of the product:
physical product 1
Enter the message of the product:
good luck
Enter the available quantity of the product:
15
Enter the price of the product:
15
Enter the weight of the product:
15
```

- The screenshot of "Display all products" function:

```
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 1
PHYSICAL - <p>
DIGITAL - <d>
PHYSICAL - <p1>
```

- Create shopping carts and add products to those two carts:

```
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 2
Enter the id of the shopping cart:
1
Enter the name of the product you want to add to the shopping cart:
p
Do you want to quit adding product to the shopping cart?(y or n):
n
Enter the name of the product you want to add to the shopping cart:
p1
Do you want to quit adding product to the shopping cart?(y or n):
n
Enter the name of the product you want to add to the shopping cart:
d
Do you want to quit adding product to the shopping cart?(y or n):
y
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 2
Enter the id of the shopping cart:
2
Enter the name of the product you want to add to the shopping cart:
p1
Do you want to quit adding product to the shopping cart?(y or n):
n
Enter the name of the product you want to add to the shopping cart:
d
Do you want to quit adding product to the shopping cart?(y or n):
y
```

- Display the new shopping carts based on their total weight:

```
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 6
[p1, d]
[p, p1, d]
```

- The cart amount of the two shopping carts:

```
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 5
Enter the id of the shopping cart:
1
The cart amount of the shopping cart is:
41.0
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 5
Enter the id of the shopping cart:
2
The cart amount of the shopping cart is:
30.5
```

- The screenshots of "Remove a product from the current shopping cart function":

```
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 4
Enter the id of the shopping cart:
1
Enter the name of the product that you want to remove:
p1
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 4
Enter the id of the shopping cart:
1
Enter the name of the product that you want to remove:
p
```

- The screenshots of all shopping carts sorted based on their total weight after removing products:

```
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 6
[d]
[p1, d]
```

- Edit physical product named p1:

```
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 3
Enter the name of the product that you want to edit:
p1
Enter the new description:
good afternoon
Enter the new avaiable quantity:
3
Enter the new price:
20
Enter the new message:
gifted product
Enter the new weight:
20
```

- The cart amount of the shopping cart whose id is 2 after editing the product named p1:
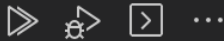
```
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 5
Enter the id of the shopping cart:
2
The cart amount of the shopping cart is:
36.0
```

- The screenshot of the "Exit" function:

```
Please enter option number 0-8:
0: Create a new product
1: Display all products
2: Create a new shopping cart and add products to the current shopping cart
3: Edit a product in the product list and the shopping cart
4: Remove a product from the current shopping cart
5: Display the cart amount
6: Display all shopping carts based on their total weight
7: Exit
Your option: 7
admin@MacBook-Pro IndividualProject %
```

- The screenshot of passed Junit tests in the project:

TESTING  ▷▷  ⇗  ⇲  ⋯

Filter (e.g. text, !exclude)  ▽

24/24 tests passed (100%)

∨ ⊘ individual_project  47ms
  ∨ ⊘ vn.edu.rmit  47ms
    ∨ ⊘ DigitalProductTest  8.0ms
      ⊘ equalSameDigitalProduct()  1.0ms
      ⊘ equalSameDigitalProductObject...
      ⊘ digitalProductCompareWithWro...
      ⊘ differentPhysicalProductName()...
      ⊘ toStringDigitalProductOutput()  1...
    ∨ ⊘ PhysicalProductTest  24ms
      ⊘ equalSamePhysicalProduct()  16...
      ⊘ equalSamePhysicalProductObje...
      ⊘ physicalProductCompareWithWr...
      ⊘ equalName()  1.0ms
      ⊘ differentPhysicalProductName()...
      ⊘ equalWeight()  1.0ms
      ⊘ toStringPhysicalProductOutput(...
    ∨ ⊘ ProductListTest  4.0ms
      ⊘ emptyProductList()  0.0ms
      ⊘ twoProductList()  2.0ms
      ⊘ sameProductNameStillCount()  1...
      ⊘ searchForProductTest()  1.0ms
    > ⊘ ShoppingCartListTest  3.0ms
    ∨ ⊘ ShoppingCartTest  8.0ms
      ⊘ testAddItemsToCart()  0.0ms
      ⊘ testRemoveItem()  1.0ms
      ⊘ testCartAmount()  2.0ms
      ⊘ testSearchForProductInCart()  2....
      ⊘ testShoppingCartOutput()  3.0ms