

Contracts

UC1: הוצאת הזמנה תקופתית מספק

createPeriodicOrder()

Operation:Cross createPeriodicOrder()

References: Periodic order from the supplier

PreConditions:

1. There are agreements in the system.

PostConditions:

1. An OrderFromSupplier instance order was created.
2. order.supplier and order.id - were initialized.
3. order.priceBeforeTotalDiscount became 0.
4. A PeriodicOrder instance periodOrder was created.
5. periodOrder.orderFromSupplier became order.
6. periodOrder.dayForInvite and periodOrder.id - were initialized.

buildProductsList()

Operation:Cross buildProductsList ()

References: Periodic order from the supplier

Preconditions:

1. OrderFromSupplier instance order was created (without products).
2. PeriodicOrder instance periodOrder was created.

PostConditions:

1. order.productsInOrder filled with products.
2. periodOrder was associated with supplier.

UC2: הוצאת הזמנה מספק עקב חוסר

CreateOrderReport(String reporter)

Operation:Cross CreateOrderReport(reporter : String)

References: Order from supplier due to shortage

Preconditions: None.

PostConditions:

1. An OrderReport instance orderReport was created.
2. Attributes of the order report were initialized.
3. orderReport was associated with reportController.

makeOrderForLastReport()

Operation:Cross makeOrderForLastReport()

References: Order from supplier due to shortage

Preconditions:

1. reportController.lastReport exist in the system
2. reportController.lastReport.barcodes not empty.

PostConditions:

1. OrderReport r became reportController.lastReport

makeOrderFromOrderReport(OrderReport orderReport)

Operation:Cross makeOrderFromOrderReport(orderReport : OrderReport)

References: Order from supplier due to shortage

Preconditions:

1. orderReport.barcodes and orderReport.amounts are not empty.

PostConditions:

1. Static A List<GenericProduct> instance productsInOrder was created by orderReport.barcodes.
2. Static A List<Integer> instance ProductsQuantity became orderReport.amounts

createProductsSupplierArray()

Operation:Cross createProductsSupplierArray()

References: Order from supplier due to shortage

Preconditions:

1. There are suppliers in the system.

PostConditions:

1. Static A List< SupplierProduct>[] instance productsSuppliersArray was created (provide products in minimum time and minimum split in the reservation)
2. Static mustBuyOrders attribute initialize (supplier that I must buy from them).

generateCombinations(int[] start, int[] end, int[] current, int index)

Operation:Cross generateCombinations(start : int[], end : int[] , current : int[], index : int)

References: Order from supplier due to shortage

Preconditions:

1. productsSuppliersArray was created
2. Static mustBuyOrders attribute initialize.

PostConditions:

1. Static Map<Supplier, OrderFromSupplier> bestSuppliersCombination was created (The optimal order).
2. bestSuppliersCombination was associated with suppliers.