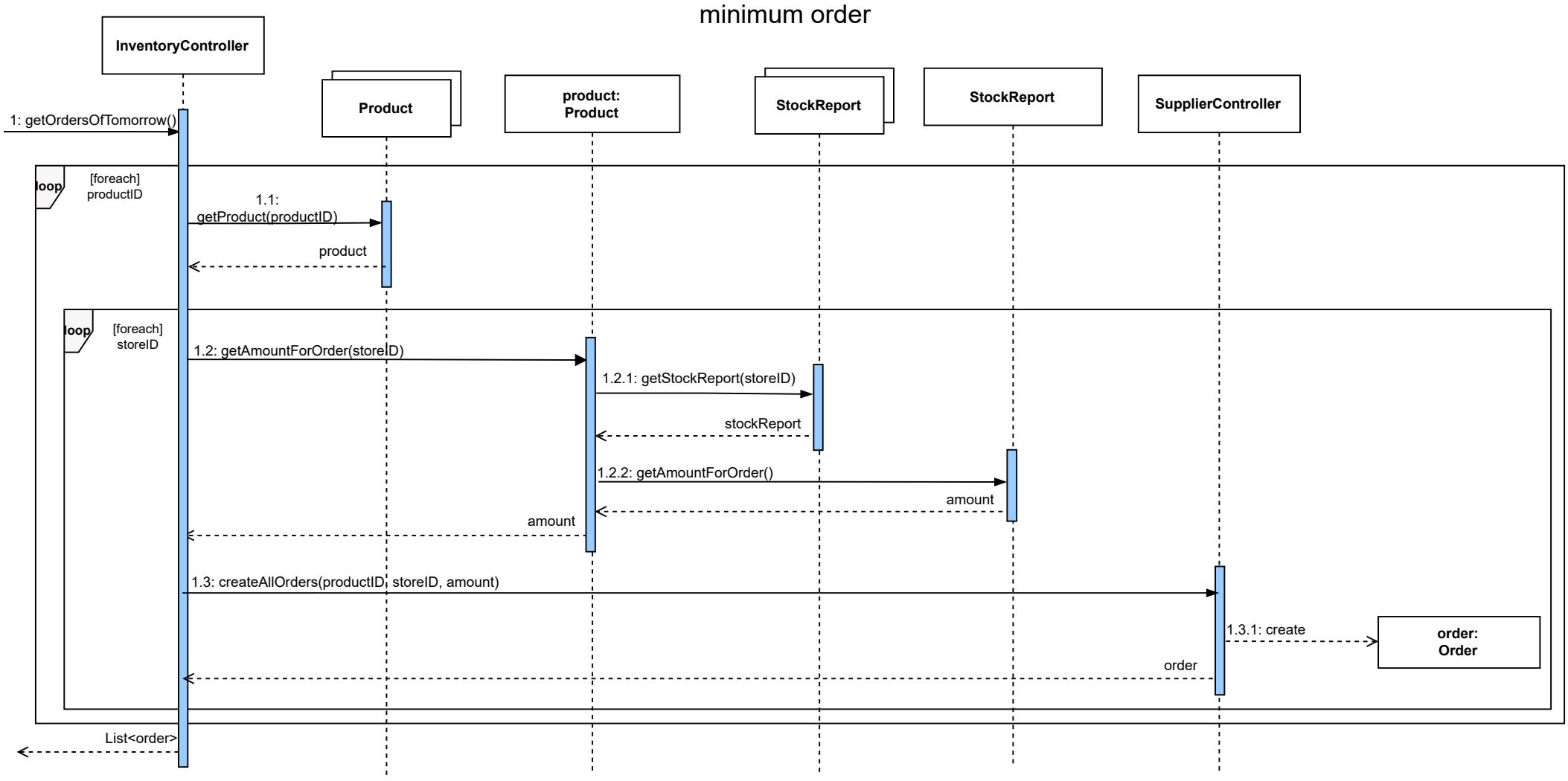


Use case name: Issuing an order from supplier due to lack of product in a specific store.

Textual description: The shift manager start the system at the beggining of each day, then all the orders of products under minimum amount are being issued to their corresponding supplier (printed on screen).

List of actors: supply manager/shift manager (the one that should see the orders report on the screen every morning).

Pre-conditions: The system must load the following data: suppliers and products

Post-conditions: The amount of the product decreased in the store (new amount=old amount-required amount<minimum amount => new order has been created and printed on screen). amountInDeliveries.contains(orderID, required amount)=true, in the corresponding stockReport.

Name: Orders On The Way

Textual description: this use case should be initiated once in a day, when the system is started and it returns to the inventory department all the orders that are scheduled for tomorrow, and all the orders requested by the inventory department due to shortage in certain items.

List of Actors: Storekeeper.

Pre-Conditions: requested items must be delivered by least one supplier.

Post-Conditions: inventory items on-the-way quantities are updated according to the orders and shortages are deleted.

Main success scenario (Pseudo-code):

User

1) Start the system at the beginning of the work day.
2) System orders automatically from suppliers.
3) The new orders are available to watch in Supplier System.

Inventory Model

startOfDay:

1) Find the items to order, the requested amount for each one and what store request this order.
2) Ask supplier model to order requested items and routine orders (call function createAllOrders).
3) Update all products amount in inventory.

Supplier Model

createAllOrders(mapOfShorage):

1) Declare List<Order> *finalOrders*.
2) Declare List<Order> *autoOrders*.
3) Load all existing routine orders from Data Base, add all to *finalOrders*.
4) For every routine supplier:
   a) If the supplier has a scheduled delivery for tomorrow, order from him the last order.
   b) Add order to *autoOrder*.
5) For every item in mapOfShortage:
   a) Search for the item in all the orders in *autoOrders*, and subtract the quantity ordered of this item to the requested store.
   b) Add any order contains the item to *finalOrders*, remove it from *autoOrders*.
   c) Update requested quantity of the item in mapOfShortage d) If the requested quantity for the item is 0, delete item from mapOfShortage.
6) For each item in mapOfShortage:
   a) Search for supplier that sell this item with the cheapest price. b) If there is an order from the supplier to the requested store in *finalOrders*: update the order to contain the item in the wished quantity. Else: If there is an order in *autoOrder*s from supplier, delete the order and make a new order for the item, add it to *finalOrders*. Else: make a new order for the item, add it to *finalOrders*.
7) Add all orders from *autoOrder*s to *finalOrders*.
8) Return *finalOrders*.

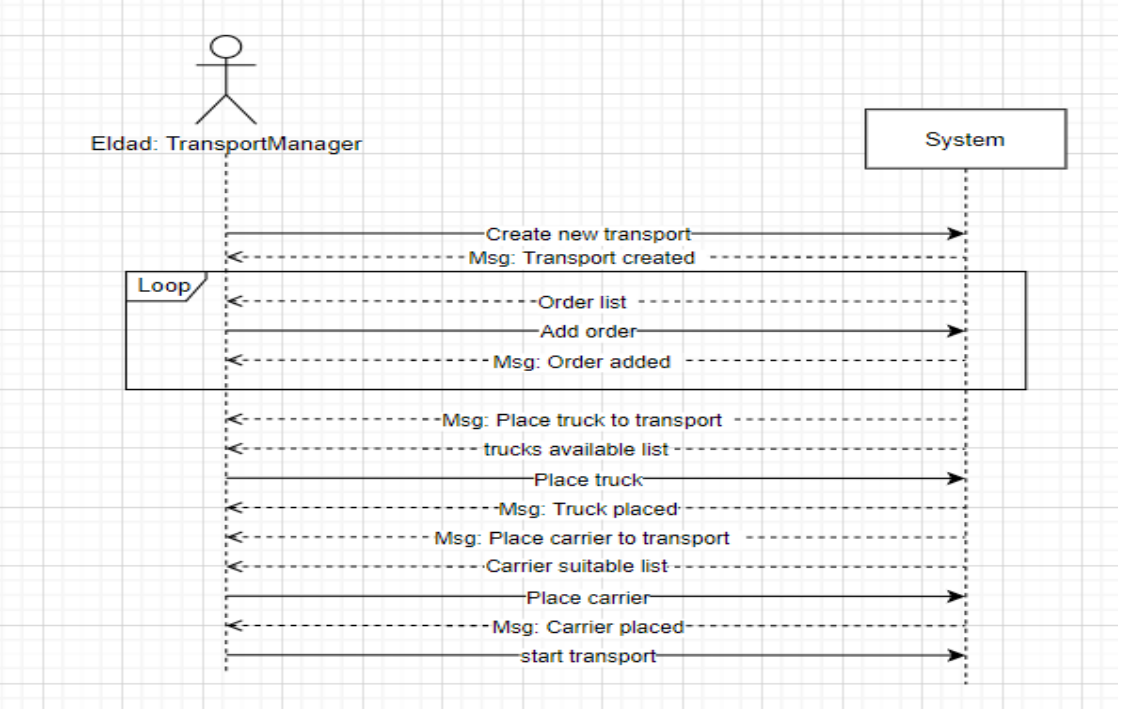
Alternative\Extantions:

a) System failure when initiated:
   1) System will present the error message to the user that booted the system.
   2) System will ask the user to restart it.

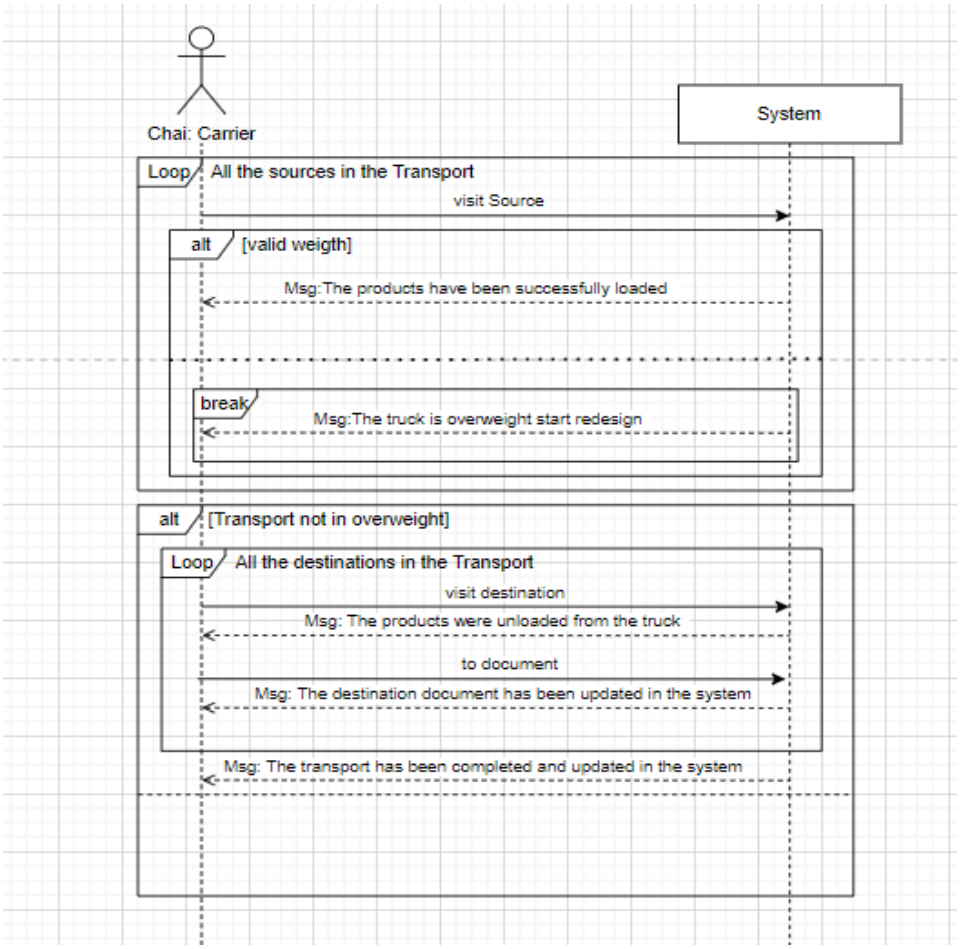
:Use case

Use case name	Carrying out its transport
Textual Description	Creating and carrying out its transport.
List of Actors	Transport manager and Carrier.
Pre-conditions	1. There is an order for transportation. 2. There is a truck and a carrier with a suitable license for this truck. 3. There are sources and destinations of transportation.
Post-conditions	1. In the completed transports archive the order exist. 2. The truck and driver will be available for other transports.
Main success scenario	1. Transport Manger create new transport. 2. Transport manager adds transport order to the transport. 3. Transport manager chooses truck for the transport. 4. Transport manager chooses driver that suitable to this transport. 5. The carrier updates the truck weight in each source. 6. The carrier updates about his visit in each destination, and the destination document will save in the archive. 7. When the carrier finishes his ride the transport will finish, and transport document will save in the archive.
Alternatives/Extensions	In case that the truck is in overweight the alert will send to the carrier and the redesign of the transport will be performed. The transport will end and start over. In the redesign of the transport the system will operate to reduce the weight of the truck so the transport will not be in over weight again.

sequence diagram for Transport manager with the system



sequence diagram for Transport manager with the system



Detailed Use Cases:

Use case name	Assigning-Employee-to-Shift
Textual Description	The HR manager assigns a specific employee to a specific shift.
List of Actor	HR Manager
Pre-Conditions	The HR manager is in the ShiftMenu, The shift must exist in the system; The employee has to have a constraint matching shift date and type; The employee must not be assigned to the shift
Post-Conditions	The shift has the employee assigned to it; the database includes a record which details the assignment
Main success scenario	1) The HR manager chooses the shift through the ShiftMenu 2) The HR manager chooses to add to the shift employees of the type of the wanted employee 3) System validates number of already assigned employees of chosen type is less than set max number for this type 4) System shows available employees for chosen shift of chosen type 5) HR Manager chooses the employee out of the list of available employees 6) System checks new number of currently assigned to set max 7) HR Manager stops the assignment process
Alternative/Extensions	3') Number of assigned employees matches set number 4') System notifies HR manager of the situation and offers to reset the number of employees for the shift or remove already assigned employee
	7'') System stops the assignment process

When assigning an employee to a shift the system will interact with the manager in order to produce identifying data for the shift and the employee. The system will assign and save the assignment in the system if conditions are met.