Use-case 3.1: getting information.

1. Actor: User

2. Precondition: There is at least one registered shop \ product.

3. Parameters: the user finds the wanted shop \ product.

4. Actions:

* User: select product \ shop information.
* System: display the info of the selected object (product \ shop).
* Good scenario: user will see the shop info after select it the click on the info button.
* Bad scenario: info will not present to the user \ will present to the user but

It’s incorrect or incomplete information.

Use-case 3.2 : search Items.

1. Actor: User

2. Precondition: none.

3. Parameters: identifying information for the wanted product.

4. Actions:

1. User: insert the identifying information.

2. System: Search for Items that connected to the giving info.

3. The search results will appear on the screen.

* Good scenario: the system find the wanted product do to the identifying information the user as giving.
* Bad scenario: the system do not find product even there is a product that fits with the information.

Use-case 3.3: Saving products in a shopping cart.

1.Actor: User

2. Precondition: the user find the wanted product.

3. Parameters: the wanted product display on the screen.

4.Actions:

1. User select the product.

2. the system update the user cart if the product is in stock.

* Good scenario: the system will update the product in his cart.
* Bad scenario: the system will connection to the wrong cart (other user) \ connection with the wrong product \ will no make connection at all.

USE- case 3.4 Editing the shopping cart

1. Actor: User

2. Precondition: the user have at least one Item on the cart.

3. Parameters: none, action of the user.

4.Actions: for each product will appears the quantity.

* + Remove Item:
    - User: The user open he’s cart.
    - User: edit the number to ‘0’.
    - User: save the action.
    - System: remove the item from the user cart.
  + Edit the quantity:
    - User: edit to the wanted quantity.
    - User: save the action.
    - System: check at the shop, product from if it can provide the wanted quantity.
    - System: If it’s possible, will update the shop about the quantity. Other, the system will not allow to save the user action.
* Good scenario: the product will not appears in the user cart / will appears with the new wanted quantity. System do not edit the quantity do to shop disapproval.
* Bad scenario: the product will appears with quantity ‘0’ / wrong quantity / other Item will be updated.

Use-case 3.5: check-out (buying)

1. Actor: User

2. Precondition: the user have at least one Item on the cart.

3. Parameters: none, action of the user.

4.Actions:

* + The user open he’s cart.
  + User click on the payment icon.
  + The system calculate the total amount of the product and display it on the screen.
  + The User insert the payment details
  + The System divides the payments by stores and sends the payment amount, for which store with the payment details to the external service (Payment option).
  + After receiving approves that payment done the system will send the invoicing to the associated shops and display the invoicing for the user.
  + The system will save that action in the data base, both the user and for the stores associated with the transaction.
* Good scenario: The user manages to make a payment for the products kept in the shopping cart, each store receives an invoice for its part in the transaction.
* Bad scenario: The system is unable to make the payment \ the stores do not receive notification of the transaction \ A payment is made that does not match the user's shopping cart.