All need to check login information in **logService**.

**Level**:

IDatabaseManipulationImpl

Service

Dao

Trigger

**State**:

Picking-up

To-Export Transporting

Export Checking

Export Check Fail

Packing to Container

Waiting for Shipping

Shipping

Unpacking from Container

Importing Checking

Import Check Fail

From-Import Transporting

Delivering

Finish

1. newItem
   1. state: Picking-up
   2. check:
      1. IDatabaseManipulationImpl: itemInfo retrieval city = courier city
      2. IDatabaseManipulationImpl: some info need to be **null**
   3. operation:
      1. insert
2. setItemState
   1. state: Picking-up -> To-Export Transporting
   2. check:
      1. Service: logInfo.id=retrieval courier id
      2. Trigger: old state=Picking-up
   3. Operation :
      1. Update state
3. setItemState
   1. state: To-Export Transporting -> Export Checking
   2. check:
      1. Service: logInfo.id=retrieval courier id
      2. Trigger: old state= To-Export Transporting
   3. Operation:
      1. Update state
4. setItemCheckState
   1. state: Export Checking-> Export Check Fail
   2. check:
      1. IDatabaseManipulationImpl: delivery export city=seaport officer city
      2. Trigger: old state= Export Checking
   3. Operation:
      1. Update export\_officer\_id, state
5. setItemCheckState
   1. state: Export Checking-> Packing to Container
   2. check:
      1. IDatabaseManipulationImpl: delivery export city=seaport officer city
      2. Trigger: old state= Export Checking
   3. operation:
      1. update export\_officer\_id, state
6. loadItemToContainer
   1. state: Packing to Container-> Waiting for Shipping
   2. check:
      1. IDatabaseManipulationImpl and trigger: container: using = false
      2. trigger: delivery: old state= Packing to Container
   3. operation:
      1. update delivery set container\_code, state
      2. trigger: insert into sea transportation
      3. trigger: update container set using = true
7. loadContainerToShip
   1. state: Waiting for Shipping -> Waiting for Shipping (no change)
   2. check:
      1. IDatabaseManipulationImpl: ship: sailing=false
      2. IDatabaseManipulationImpl: container: using=true?
      3. Trigger: sea transportation: container exists
      4. Trigger: sea transportation: ship=null
   3. operation:
      1. update sea transportation set ship
      2. trigger: update delivery set ship\_name
8. shipStartSailing
   1. state: Waiting for Shipping -> Shipping
   2. check:
      1. IDatabaseManipulationImpl and trigger: ship: sailing=false
   3. operation:
      1. ship: update sailing=true;
      2. trigger: update delivery (in this ship) set state
9. unloadItem
   1. state: Shipping -> Unpacking from Container
   2. check:
      1. IDatabaseManipulationImpl: sea transportation: ship (of this item) is not null
   3. operation:
      1. sea transportation: delete all records of this ship
      2. trigger: ship: sailing=false
      3. trigger: container: using=false
      4. trigger: delivery: state= Unpacking from Container
10. itemWaitForChecking
    1. state: Unpacking from Container -> Importing Checking
    2. check:
       1. trigger: delivery: old state = Unpacking from Container
    3. operation:
       1. update delivery set state = Importing Checking
11. setItemCheckState
    1. state: Import Checking-> Import Check Fail
    2. check:
       1. IDatabaseManipulationImpl: delivery: delivery import city = seaport officer city
       2. Trigger: delivery: old state = Import Checking
    3. Operation:
       1. Update import\_officer\_id, state
12. setItemCheckState
    1. state: Import Checking -> From-Import Transporting
    2. check:
       1. IDatabaseManipulationImpl: delivery: delivery import city = seaport officer city
       2. Trigger: delivery: old state = Import Checking
    3. Operation:
       1. Update import\_officer\_id, state
13. setItemState
    1. state: From-Import Transporting -> From-Import Transporting (no change)
    2. check:
       1. IDatabaseManipulationImpl: delivery: delivery city = logInfo city
       2. trigger: old state = From-Import Transporting
       3. trigger: old delivery courier id = null
    3. Operation:
       1. Delivery: update delivery courier id, state
14. setItemState
    1. state: From-Import Transporting -> Delivering
    2. check:
       1. Service: logInfo.id = delivery courier id
       2. trigger: old state = From-Import Transporting
    3. Operation:
       1. Update state
15. setItemState
    1. state: Delivering -> Finish
    2. check:
       1. Service: logInfo.id = delivery courier id
       2. trigger: old state= Delivering
    3. Operation:
       1. Update state