# A/ Use-cases: A-Sell item

## A1/ Post a sell record – user doesn’t have their own private catalogs

### Use-case description

* User login
* Select a catalog (A) from the system catalog tree

*API: PartnerManager.getSystemRootCatalog()*

*PartnerManager.getSystemSubCatalogs(parentSyscatId)*

* Present user the list of items in that system catalog

*API: PartnerManager. getSyscatItemPagingList (syscatId)*

* User select an item to sell:
  + If an item is found, move to next step
  + If no item is found, show user new item form. User will fill in all details of the item, such as name, description, branch, and other fields, and:
    - *CatalogItem.setSyscatId(A)*
* Filling price, and possibly: quantity, sell message

*API: missing*

### Core API for use-case

## A2/ Post a sell record – user has their own private catalog

### Use-case description

* User login
* Select a catalog (A) from the their private catalog tree

*API: user.getBranch().getRootCatalog()*

* Present user the list of items. Based on the type of the system catalog that associated to the user’s selected private catalog, this list could be:
  + The list of items in the system catalog, if the system catalog is unmanaged

*API: PartnerManager. getSyscatItemPagingList (syscatId)*

* + The list of items in user’s selected private catalog, if the catalog is managed

*API: Catalog.getCatalogItemPagingList()*

*API: Catalog.getSystemCatalog().getCatalogSpec().isManaged()*

* User select an item to sell:
  + If an item is found, move to next step
  + If no item is found, show user new item form. User will fill in all details of the item, such as name, description, branch, and other fields, and:
    - Select a catalog (B) from the system catalog tree (starting from the system catalog associated to (A)

*API: PartnerManager.getSystemSubCatalogs(parentSyscatId)*

*CatalogItem.setSyscatId(B)*

* Filling price, and possibly: quantity, sell message

*API: missing*