Service Layer

StoreDTO
-

<<interface>> IStoreService
-

ServiceFactory
-

<<interface>> IUserService
-

UserDTO
-

ProductDTO
-

StoreService
-

IUserService
-

- returns
- creates
- creates
- returns
- 1
- 0..*

Domain Layer

Market Package

Store Package

StoreController

PurchasePolicy

DiscountPolicy

Store
- storeID
- isOpen

Discount

VisibleDiscount | HiddenDiscount

Inventory

Product
- storeID
...

has a
has a

User Package

UserController

NotificationSystem

User

Subscriber
- username

Guest

Owner

Admin

Manager

ManagerAuthorization <<enum>>
- View
- UpdateSupply
- UpdatePurchasePolicy
- UpdateDiscountPolicy

ShoppingCart

ShoopingBasket
- storeID

Order
- isComplete

founded by
appointed
notifies
notifies
owned by
managed by
appointed
created from

Note Example

Each registered user (subscriber) has a unique username

System Administrator (admin) is a kind of subscriber with managing permissions, there has to be at least 1 at all times

Each shopping basket contains the requested products from a single store

Represents the order history of the user, each order is created from the current shopping cart, copies all information from the baskets and checks the availability using the external services before being marked as complete.
If one of the external services failed the order is marked as failed but could still be saved in the order history

External Services

SupplierService

PaymentService

-uses-
uses