Simple Basket API and Client

Project has been implemented according Clean Architecture.

Project can be downloaded from https://github.com/krmdrsvgl/myshoppingbasket

Tests

In this folder there are various test covering fundamental parts of the application and its dependencies. As project evolves, many more tests can be added.

Xunit, Mocq are libraries are used in test implementations.

Shopping Basket Api.

It is implemented in Asp.Net Core 2.0.5.
It can easily be tested at http://localhost:5001/swagger . It can be run from VS or console and does not require any database. All repositories and services are in memory constructs. To prevent data loss, some services are injected as singleton instances. IOC principle is used, so in future services and repositories can easily be replaced.

Application assumes that there is legit JWT token present. In future, some OAUTH2 or OPENID implementation can be added to insure security.

There are two api end points:

http://localhost:5001/api/basket :
This is the main REST API endpoint that implements adding, removing items, changing quantity, clearing out basket items. Each user has only one basket.

GET http://localhost:5001/api/basket get current basket or create one. It ensures that there is always one basket for each user.

POST http://localhost:5001/api/basket add item to basket. If item already exists, only the quantity of the existing item will be incremented.

PUT http://localhost:5001/api/basket/changeQuantity only changes the quantity of the related item.

DELETE http://localhost:5001/api/basket/delete remove the item from basket.

DELETE http://localhost:5001/api/basket/clearall remove all the items from basket.

http://localhost:5001/api/catalogitems Since it is Catalog Items is another context, Catalog Item api is only for test purposes. When catalog service is served by the other teams, it can easily be replaced.

CLIENT LIBRARY

Library is plain library. It requires the SecretKey of consumer.

Library assumes it has already acquired a JWT token of the user. To do so, an Identity Server or a third party identity could be used. Library implements all Basket APIs. In every request, secret and user`s JWT token are sent to API endpoint.