SDV701

Assignment Two

Stage 1

Oleg Sivers

2020

Contents

Business scenario

Our company is going to sale network equipment and PC parts related to networking. According to requirements we should provide for the company a management tool to modify what's in stock and check orders and a frontend application for clients, where order placement will occur.

Sample list of items:

- Network interface card
- Network switch
- Wireless router
- Wired router

All devices or parts within the system will have this common fields: PartsID, Name, CategodyID, Currency, Price, QtyInStock, LastModified.

Parts are going to be categorized on three categories:

- Wired
- Wireless
- Wired and Wireless

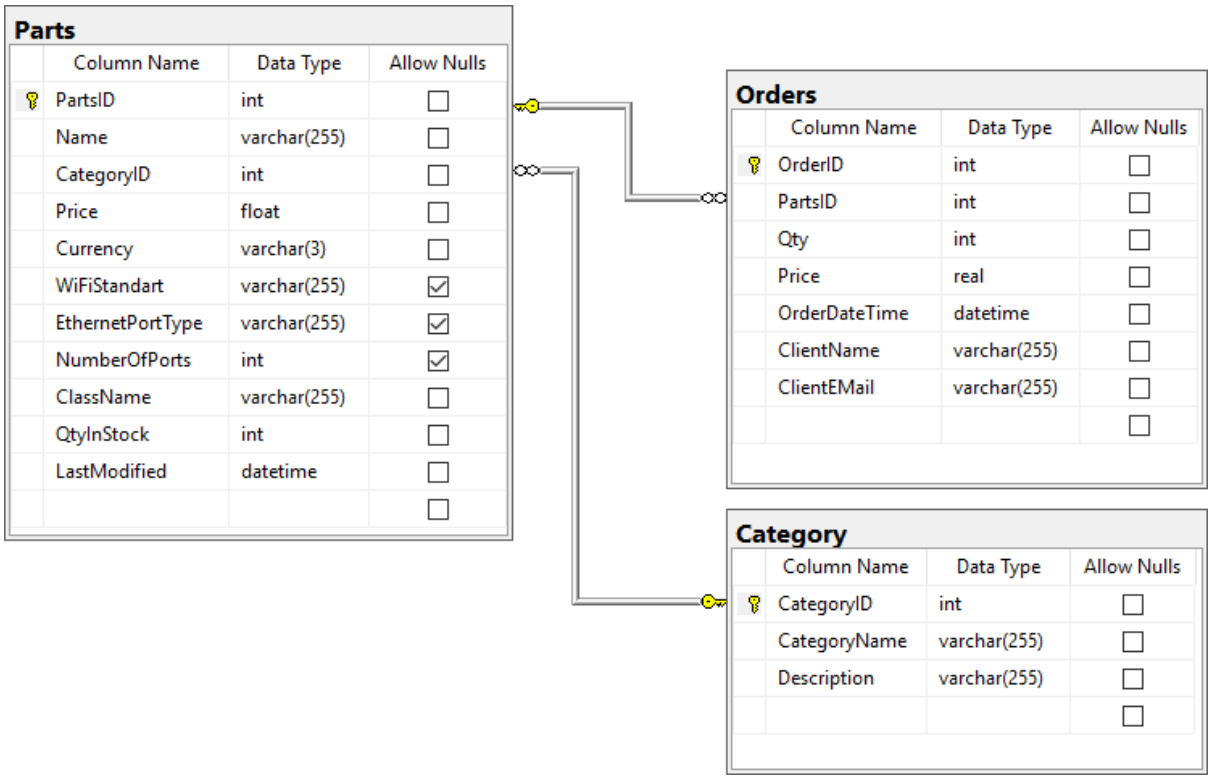
Each category will include category name and description fields.

There are extra parameters extends common parameters of parts, depending of category:

- Wired devices have extra parameters: Number of ports and port speed (10/100/1000)
- Wireless devices extra parameter is wireless standard (a\b\g\n\ac\6)
- Combined devices are having all extra parameters of wired and wireless devices

Database Schema

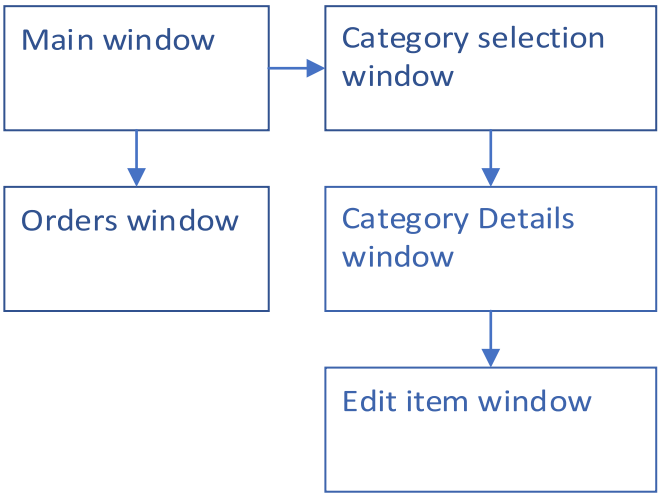
I'm going to use single table inheritance in my project so Parts table fields will include all the data for all classes in hierarchy (NPart, NWiredPart, NWirelessPart and NWiredWirelessPart) there would be an extra field with a class name (ClassName) to restore table data according to inheritance.



Applications

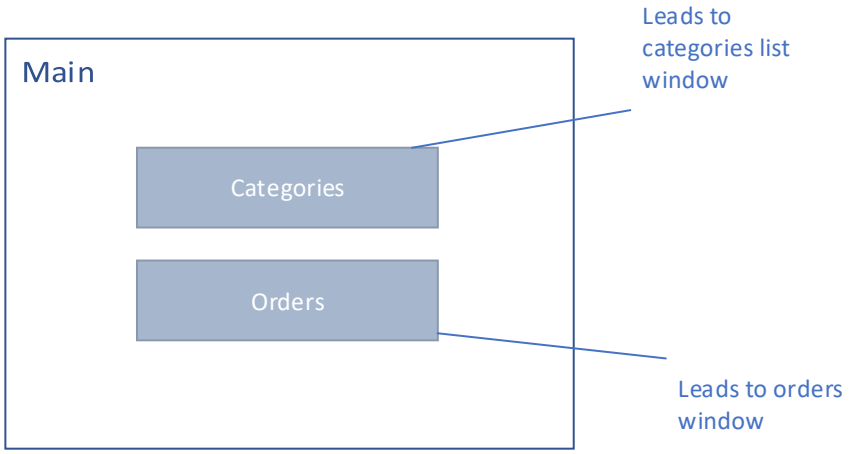
Windows WPF Client

Navigation



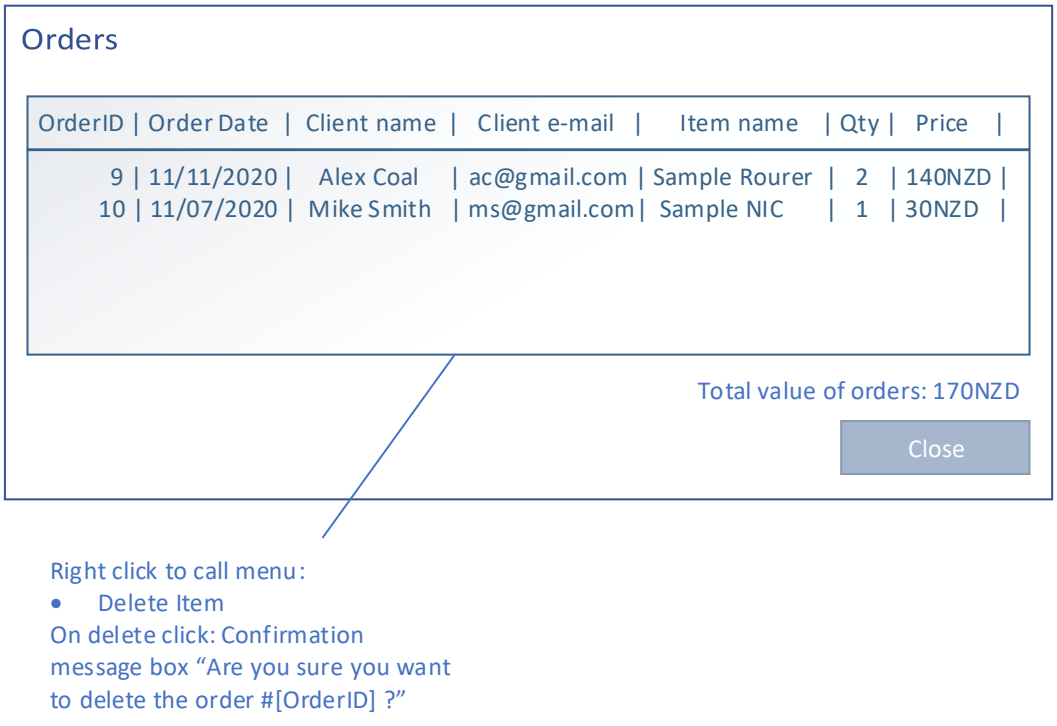
Main window

You can navigate to categories window or to orders window from main form. Clicking on both buttons will open a new window.



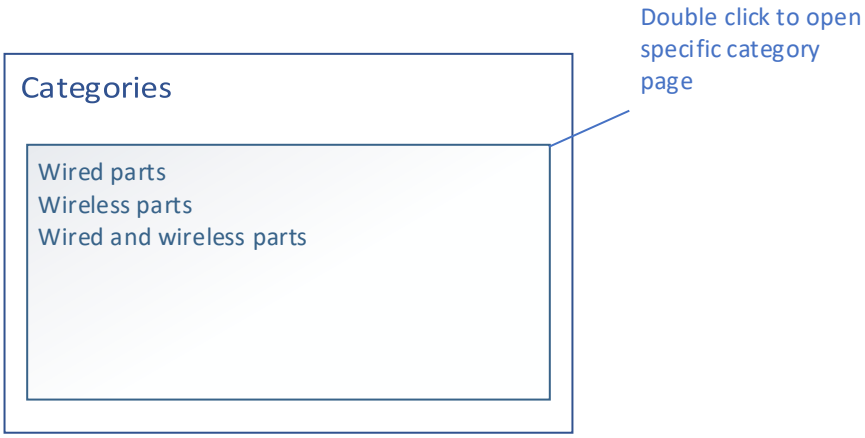
Orders window

From orders form you can use context menu to delete current selected order. A confirmation message will appear before an actual delete process. Upon receiving the confirmation application will call API to delete the order.



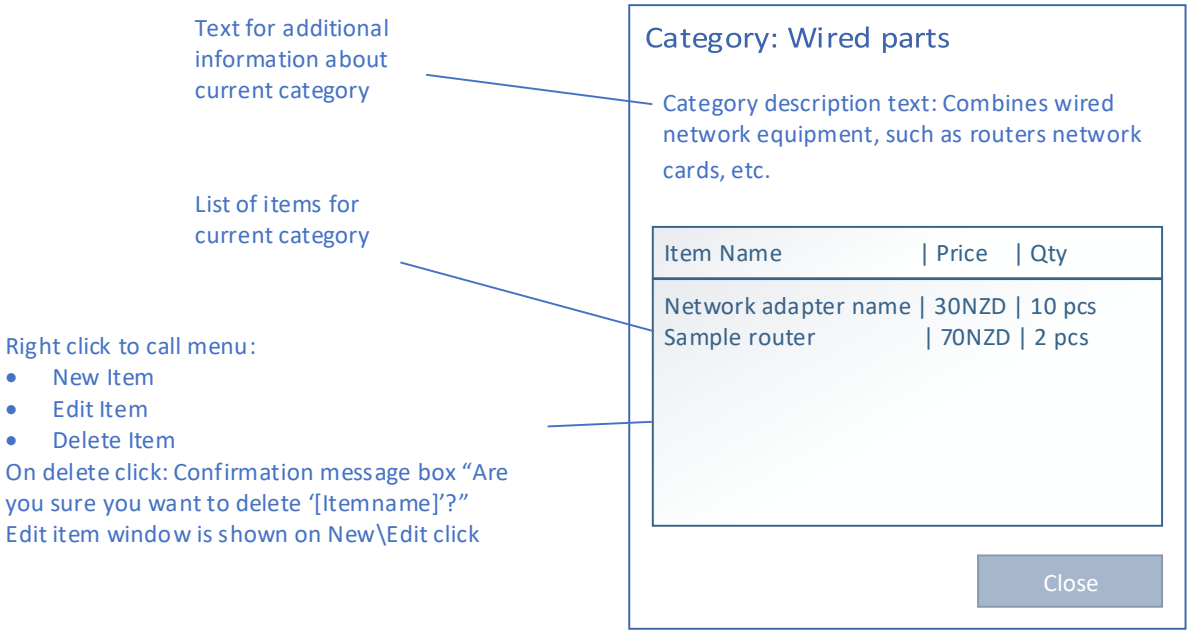
Category selection window

Shows an ordered list of categories. Clicking on category will lead to open of category details window.



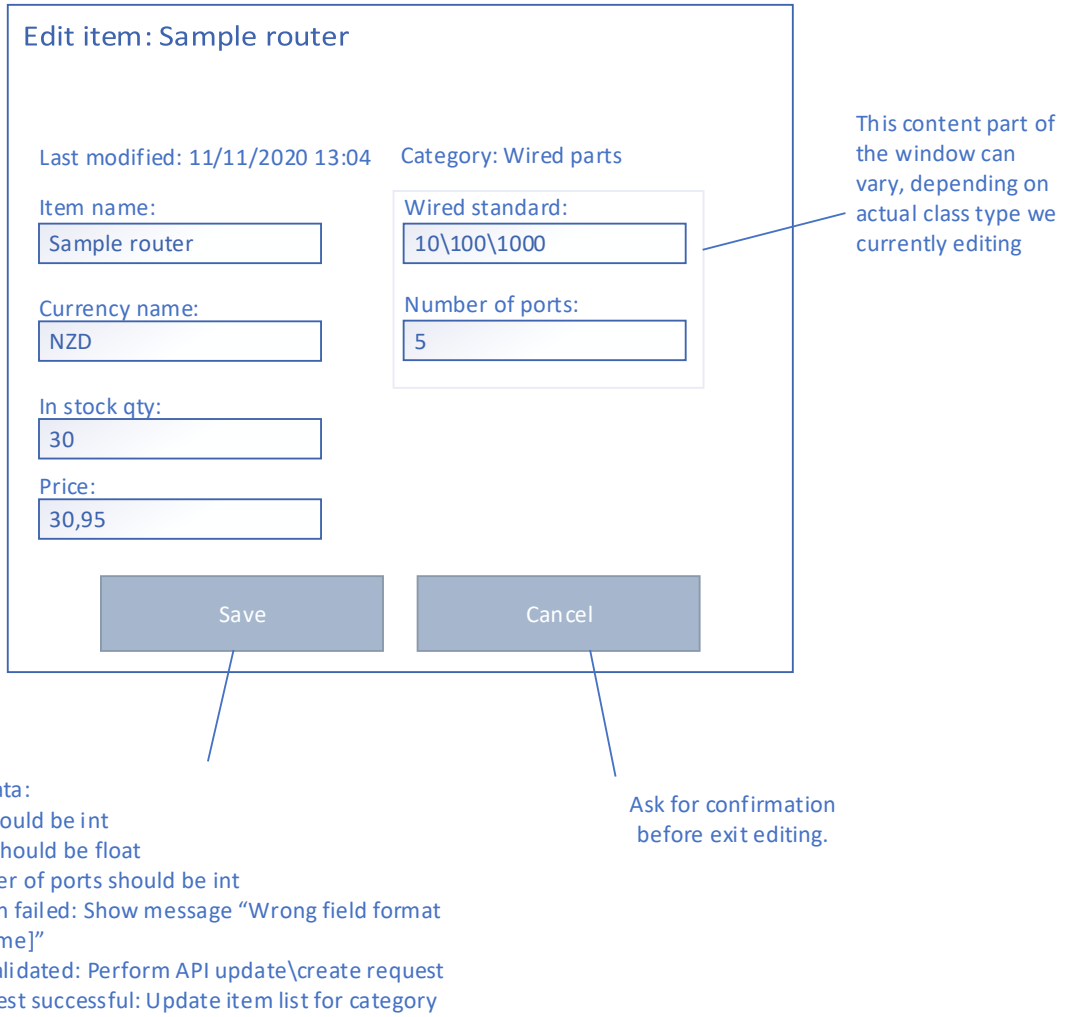
Category details window

Shows a sorted list of items, belong to current category. The list is having a context menu to create\edit\delete an item. A confirmation is requested on delete menu item click. Clicking on edit\new menu item will lead to open of Parts base window (edit item window).



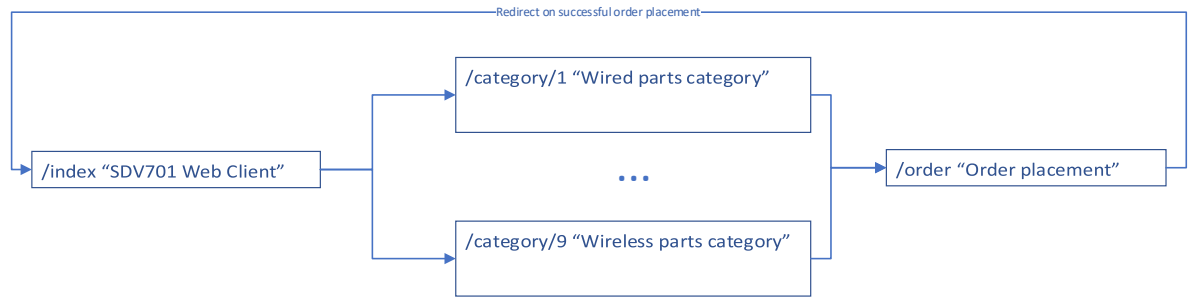
Parts base window (edit item window)

Allow to edit information about an item, window layout depends on item type so some extra fields can appear and some fields may be missing, current layout corresponds to an item from “wired parts” category. Save button click will lead to validation process, then to the API call for create\update the item. Cancel button click will lead to confirmation message shows up, either user wants to quit without saving or not.



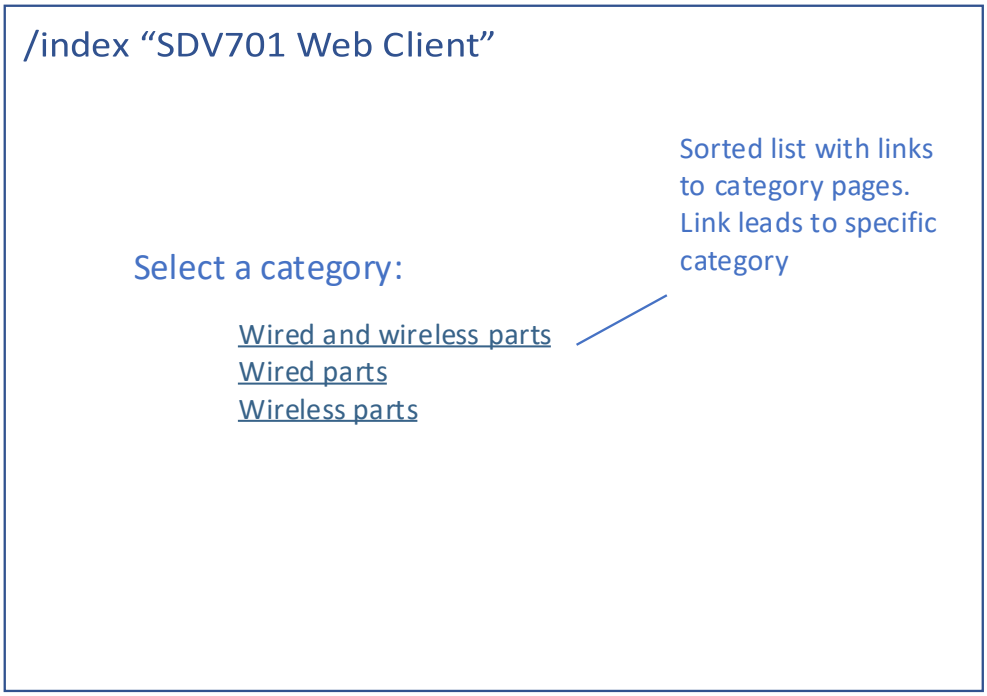
Web Client

Navigation



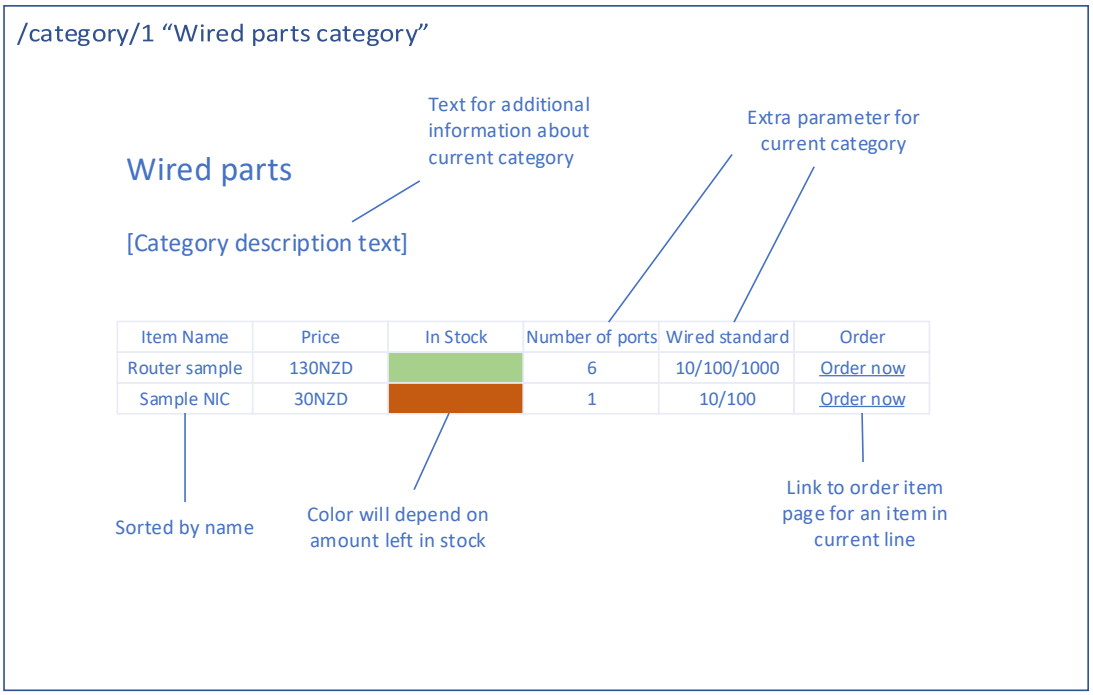
Index page

Displays an ordered list of categories. Clicking on category will lead to open of a category details page.



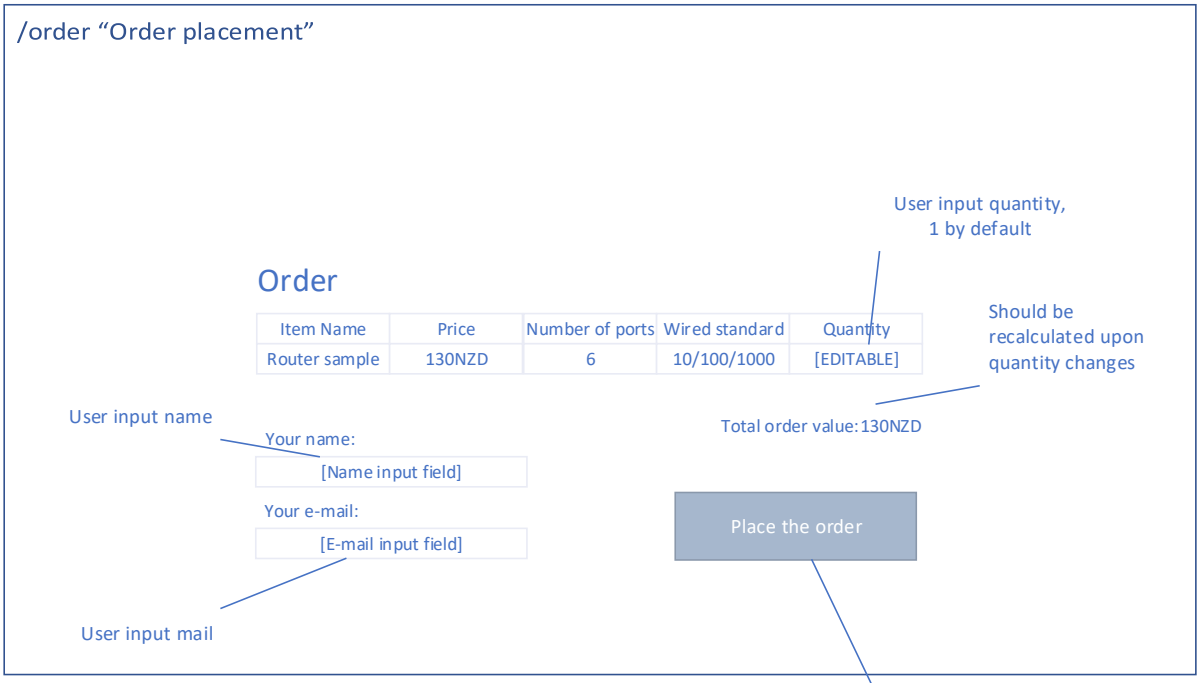
Category detail page

Displays a sorted list of items, which belongs to current category with additional info like price, in stock quantity color, item specific fields. Following “Order now” link will lead user to an “Order placement” page



Order placement page

Will allow user to input contact details, specify quantity if required and check item name and final price. Place order button will lead to data validation process followed by checking in stock quantity. On successful completion user will see an “Order confirmed message”. If the process fails user will be notified.

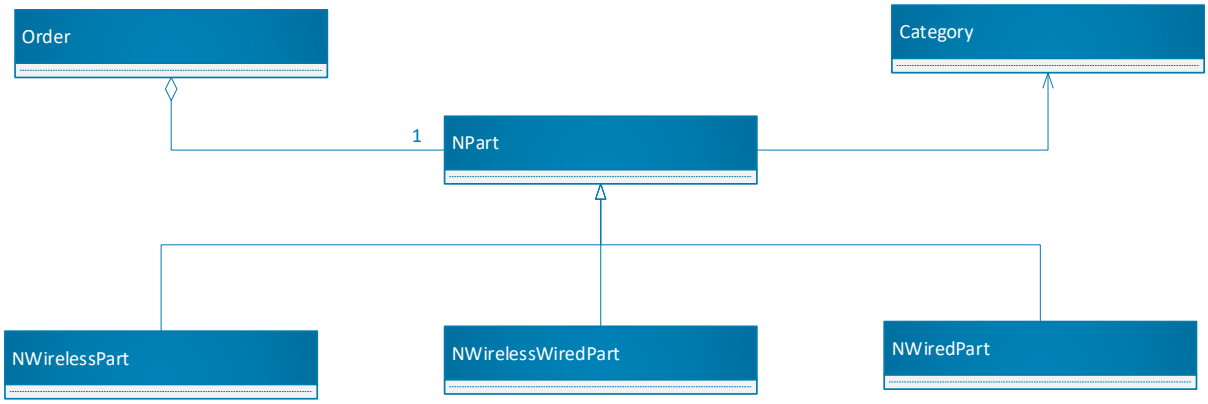


- Validate quantity and email. Highlight field red in case validation not passed.
- Check available quantity in stock(according last obtained info from DB), inform user with popup if there is not enough items in stock.
- Call stored procedure to place order (check in stock quantity again just before inserting a record)
- Display a result of stored procedure, either order is placed or not with popup.
- If order is placed redirect to index page.

UML

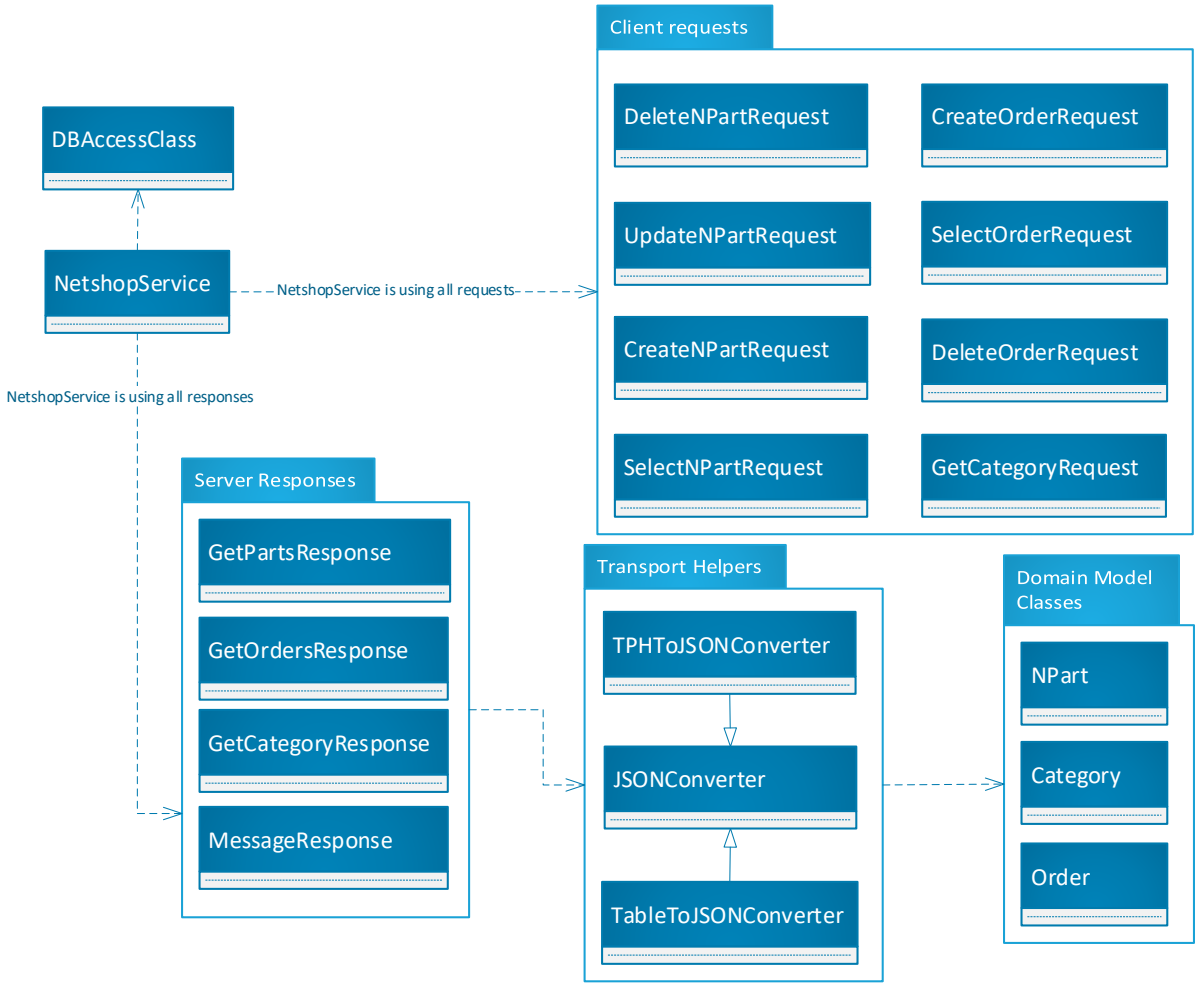
Domain model classes

Classes are placed in shared assembly. So, all applications can access it.



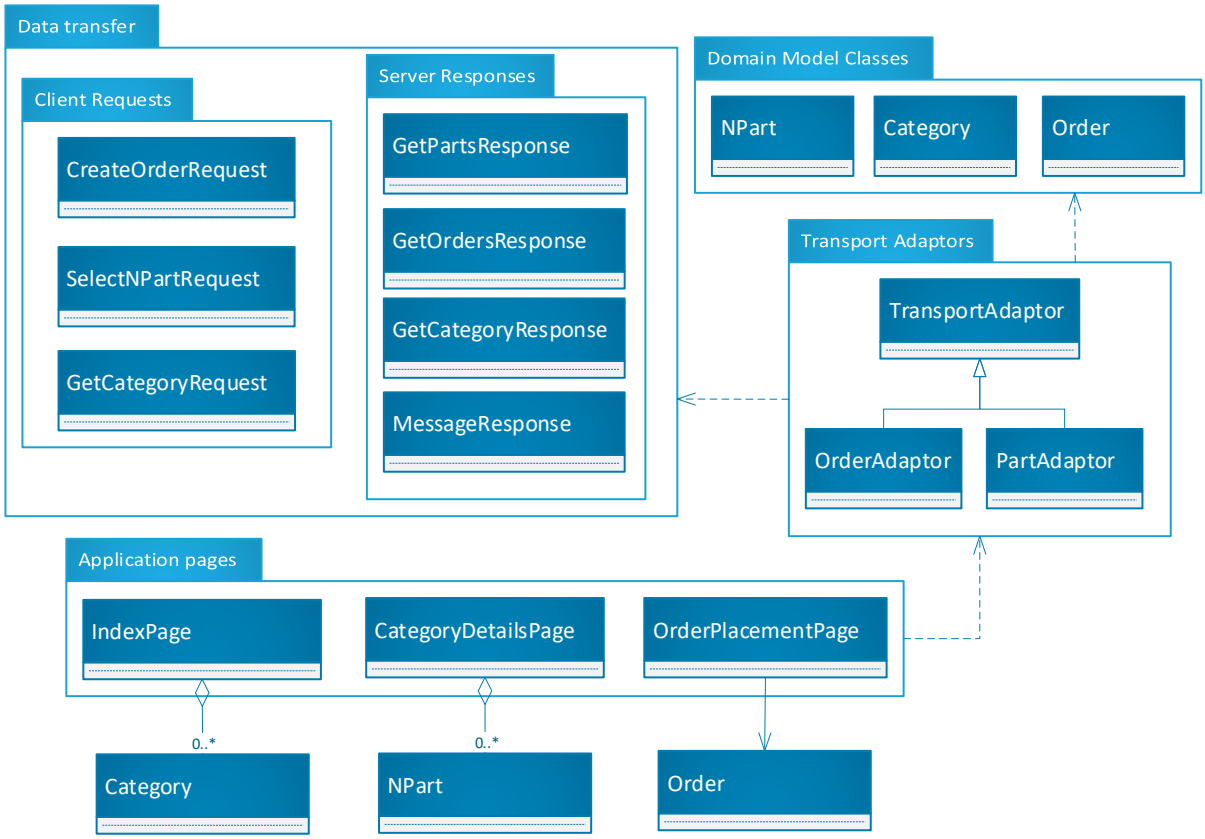
Server

For each situation server is having gRPC response, request and NetshopService method to handle request and responses transport. Each request leads to generating response ether containing a serialized set of domain model classes or a message response. Requests and respond are serving as data transfer objects, domain model classes are used to unify packing and unpacking process.



Web client

Client requests and server responses are DTOs, which carrying requested and responded data, if system requires converting of domain model class into DTO Transport adaptors are used in that case. Adaptors are also used to generate requests for create, update, delete operations.



Windows WPF client

Is using the same data transfer technic as web client, but set of requests is different.

As client is based on WPF we are using strategy pattern to replace parts of window to cover inheritance representation while displaying items in “PartsBaseWindow”, as visual inheritance cannot be done in WPF the same way as in Windows forms applications.