Peter's Cookies Ltd.

Peter has been selling his cookies on the market for years with a cookie stall. He sells ordinary, sugar-free and super cookies which are ordered from three suppliers: A, B and C.
A year ago Peter asked his nephew - who previously made a website for him - to automate the ordering process.
They have negotiated with the suppliers, agreed on an interface, and his nephew made an API for Peter to get prices and place orders. The website/UI that consumes this API is out of scope.
Peter is using the system to his full satisfaction and has great ideas for the future.
He now wishes to also to let customers order cookies through the website and for that the API needs to be extended. His nephew has already made a piece of software (see PetersCookies.sln), but it's not completely finished and he has no time left to finish it. Thus Peter is looking for someone who can deliver a working system.

Requirements

- For the first version customers don't need to have an account with Peter, providing a name and a pickup date is sufficient. Orders cannot be collected on Sundays and public holidays.
- The solution must select the cheapest possible order that meets the customer's request. This means that when the customer orders multiple types of cookies, the different types (with different delivery dates) can be ordered from different suppliers.
- Peter wants to earn 1 euro on every pack of cookies he sells.

- If the cookies come from different suppliers, the delivery dates should not be more than 1 day apart.
- Supplier A calculates 5 euro shipping costs and delivers 4 business days after ordering.
- Supplier B charges 5 euro shipping costs, but these will be zero when orders exceed 50 euros. Supplier B delivers 3 business days after ordering. Supplier B can not be ordered on Sundays and National Holidays.
- Supplier C is an American company, delivering 5 days after ordering and calculates 5% shipping.

Interactive service specifications of supplier APIs

- Supplier A: http://stroopwafela.azurewebsites.net/swagger
- Supplier B: http://stroopwafelb.azurewebsites.net/swagger
- Supplier C: http://stroopwafelc.azurewebsites.net/swagger

Acceptance criteria

- Peter does not understand the design and implementation of software. The software that was written by his nephew is therefore very flawed at various points. Since Peter has more future plans for the site, we would like to see you improve, clean up and expand the software. In addition to the functional correctness, we are also looking at things like code structure and design principles.
- The code of the solution must be production ready (except the "Outside scope").
- The solution should be delivered as a Visual Studio solution (in zip), excluding any packages from nuget.org. Please also remove the bin and obj folders.
- Please help Peter checking your solution by providing some examples for requests.
- If a database is used, it should be usable for us without further instructions.

Out of scope

- Client application / UI
- Security mechanisms
- REST services from suppliers (these are available online)
- Integration, acceptance and chain tests: these are developed in a later phase

Notes

- The time spent on this assignment should be no more than 3 to 5 hours (to not overload you).
- To stay within that time, it is highly recommended to continue with what is available and make an MVP from that! The business logic around the wish date / delivery data is the most complex so leave it for the end.
- If you want to stop at some point and can’t finish it all, just send us what you have.