Paulo Monis
Candidate

paulo.monis.it.consulting@hotmail.com
+31 6 12 78 06 10
Den Haag, Netherlands
linkedin.com/in/paulo-monis-b113a6112

Hi Fedex, thank you for the opportunity of doing this assignment.

Please find below a filled in checklist and also the requested design decisions.

- AS-1 implemented - **DONE**
- AS-2 implemented - **DONE**
- AS-3 implemented - **DONE**
- README with run instructions - **DONE**
- Document with design decisions - **DONE**

Design decisions:

I believe that for this assignment the key challenges were throttling, bulking and querying multiple external points at once. And that's why I chose **WebFlux** Framework for this assignment.

Spring WebFlux is used to create fully asynchronous and non-blocking application built on event-loop execution model, which fits perfectly for the requirements of this application.

The project is divided in the following packages:

controller

This is where the requests come in and we zip all the request needed (shipment, pricing and track), and by using the method .block on the Mono resulted by the zip, we assure that the user will only receive a response once all the requests are completed.

service

The services is where all the throttling and queuing happens. A Flux queue is created and by calling bufferTimeout we will ensure that the flux is only executed once either the call cap or the time cap is achieved. Once it does, the requests are made.

At this moment each external API has it's own service but something that could be improved in the future would be making a generic code for the 3 services.

client

Finally we have the client package. Here is where we make the request for the multiple APIs using webClient. Again, we can improve here by making the code more generic and having one class instead of 3. Or at least a parent class to be able to reuse some code.