Statement

Moonhotels' HUB

Moonhotels, a new FDSA client, is a company that sells hotel room stays. Its core product is a booking engine that connects to external providers to return the availability of rooms, while also applying Moonhotels business rules.

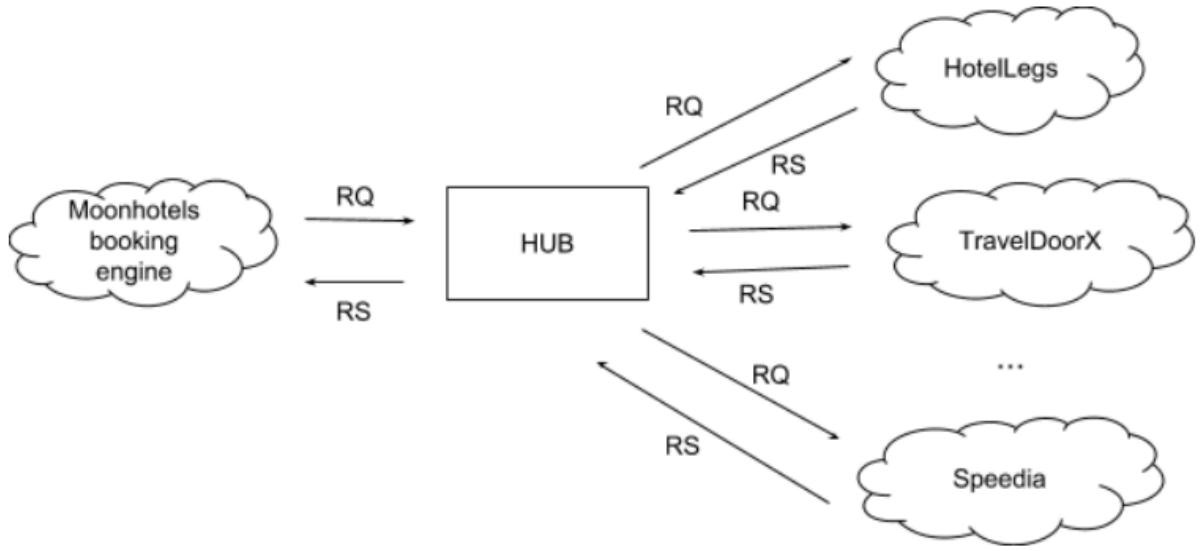
As a solution to the technical debt that has been increasing in the last years, they decided to create a new HUB that will be in charge of requesting information from the providers and aggregate the results in a unique, consolidated response.

This new HUB will only have one method, Search, that will execute the following flow:

1. Receive a search request in common format (HUB format).
2. Translate the HUB's search request to multiple search requests, one for every provider. All providers have a different search format.
3. Translate every provider response (in a different format for each one) to a response in the HUB format.
4. Aggregate the different responses in only one.

The HUB will connect to a lot of different providers: HotelLegs, TravelDoorX, Speedia, etc.

Architecture overview



Goal

Your task is to develop **the domain** of the HUB. For an initial phase of the project, there are two main tasks:

1. Create the HUB system that allows you to connect to several providers.
2. Create the HotelLegs provider integration.

As a summary, you must implement:

- The HotelLegs connector that, given a HUB Search request, calls the HotelLegs API and returns a HUB response.
- The HUB infrastructure that, given a HUB request, calls all available connectors, returns their responses and aggregates them in a single response.

Keep in mind that

- You must define a *IHotelLegsAPI* interface with the Search method, as another team member will develop the implementation that performs the real HTTP request. For this test, you can just simulate or mock it.

Requirements

- Programming language:
  - .NET or Java
  - Freedom of choice of frameworks and libraries
- Functional, executable project
- The project source code
- Business requirements stated in Goal are met

Json models

Hub Search format

Request

```javascript
{
    "hotelId": 1,
    "checkIn": "2018-10-20",
    "checkOut": "2018-10-25",
    "numberOfGuests": 3,
    "numberOfRooms": 2,
    "currency": "EUR"
```

```
}
```

Response

JavaScript


```
{
    "rooms": [
        {
            "roomId": 1,
            "rates": [
                {
                    "mealPlanId": 1,
                    "isCancellable": false,
                    "price": 123.48
                },
                {
                    "mealPlanId": 1,
                    "isCancellable": true,
                    "price": 150.00
                }
            ]
        },
        {
            "roomId": 2,
            "rates": [
                {
                    "mealPlanId": 1,
                    "isCancellable": false,
                    "price": 148.25
                },
                {
                    "mealPlanId": 2,
                    "isCancellable": false,
                    "price": 165.38
                }
            ]
        }
    ]
}
```

HotelLegs Search format

Request

JavaScript

```
{
    "hotel": 1,
    "checkInDate": "2018-10-20",
    "numberOfNights": 5,
    "guests": 3,
    "rooms": 2,
    "currency": "EUR"
}
```

Response

JavaScript

```
{
    "results": [
        {
            "room": 1,
            "meal": 1,
            "canCancel": false,
            "price": 123.48
        },
        {
            "room": 1,
            "meal": 1,
            "canCancel": true,
            "price": 150.00
        },
        {
            "room": 2,
            "meal": 1,
            "canCancel": false,
            "price": 148.25
        },
        {
            "room": 2,
            "meal": 2,
            "canCancel": false,
            "price": 165.38
        }
    ]
}
```