ERP

This application consists of several independent services. Each service is a WebAPI with 2 entities and some data manipulation methods. For each entity, a data model is introduced that should be followed to receive and store the data. Finally, each service should be dockerized and pushed to DockerHub. In the documentation page of DockerHub repository, you must have a docker-compose code that uses your service image and your chosen database image to entirely run your service. You can find a similar pattern in many DockerHub repositories. Additionally, you should push your source code to a GitHub repository and leave the repository address on DockerHub page. (2 images: app and DB(in services of docker compose you need to write these two as services and define environment variables))

In this file, a set of requirements are listed that you should try to follow them. If for any reason you think it is not viable to implement the service in the instructed way or you are unable to follow them, you can proceed with every possible scenario and then write an explanation letter of what you have done and why.

General rules

* All returned data are in JSON format
* Domain entities (Models) are the same DataTransferObjects (Service Model) //
* The type of “ID-TYPE” is a convention based on the database and programming language which is used. So please choose id types accordingly. For instance, if you use MS SQL you should use GUID and if you use Mongo you use ObjectId
* REMOVE: Services that are not the owner of the entity will only store entity ID. There are no hard relations for those properties
* We only use POST and GET, no (PUT, PATCH)
* Implementation of DELETE requests for each entity is optional
* Parameters from POST methods are in the request body and parameters from GET methods are in QueryString or URL parameter

Sales

You will develop Sale service. The sales team will manage customers and make orders on their behalf.

Entities

1. Customers
2. Orders

Customers

Examples

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id | Name | Email | PhoneNumber | Address |
| Customer-1 | Peter | peter@gmail.com | 4389836328 | Montreal, QC |
| Customer-2 | Jane | Jane@yahoo.com |  | Toronto, ON |

Methods

|  |  |  |  |
| --- | --- | --- | --- |
| URL | Type | Parameters | Description |
| /api/sale/customer/getAll | GET | (none) | Returns all records from customer entity |
| /api/sale/customer/create | POST | CustomerModel | Creates a new customer and returns created record id |
| /api/sale/customer/get/<customer-id>  E.g.,  /api/sale/customer/get/customer-1 | GET | customer-id | Returns the customer information |

Model

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Required - Default | Description |
| Id | ID-TYPE | False - New id | The unique identifier of the entity |
| Name | string | True | Customer name |
| Email | string | True | Customer email address |
| PhoneNumber | string | False | Customer phone number |
| Address | string | False | Customer shipping address |

Orders

Examples

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Id | CustomerId | OrderItems | TotalAmount | OrderDate |
| Order-1 | Customer-1 | Item1, Item 2, Item 3 | 433$ | 2/3/2023 |
| Order-2 | Customer-1 | Item 1, Item 4 | 200$ | 10/2/2023 |

Methods

|  |  |  |  |
| --- | --- | --- | --- |
| URL | Type | Parameters | Description |
| /api/sale/order/getAll | GET | (none) | Returns all records from order entity |
| /api/sale/order/create | POST | CustomerModel | Creates a new order and returns created record id |
| /api/sale/order/get/<order-id>  E.g.,  /api/sale/order/get/order-1 | GET | order-id | Returns order information |
| /api/sale/order/customer/<customer-id> | GET | customer-id | Returns all orders for selected customer id |

Model

|  |  |  |  |
| --- | --- | --- | --- |
| Name | DataType | Required - Default | Description |
| Id | ID-TYPE | False - New id | The unique identifier of entity |
| CustomerId | ID-TYPE | True | Customer Id |
| OrderItems | string | True | List of ordered products |
| TotalAmount | string | False | Total order amount |
| OrderDate | string | False - Now | Order created date |