

Contexto :

Se ha establecido un acuerdo con un nuevo proveedor al cual es necesario integrar. Dicho proveedor posee varios depósitos distribuidos en diferentes localidades, lo cual no debería afectar a la experiencia de compra del cliente. Con el fin de garantizar esto, se requiere desarrollar un sistema que permita integrarse con el proveedor y definir cuál es la mejor opción para nuestros clientes en función de la información que este nos proporcione. Es importante destacar que el proveedor tiene otros clientes por lo cual el stock puede variar por oferta. Siempre que haya una oferta disponible debe mostrarse por lo cual por ejemplo: Si el sistema determina una oferta como la mejor pero la misma no tiene stock debe retornarse siempre que se pueda la siguiente mejor. De esta manera siempre se mostraría un oferta con stock.

Contexto Técnico:

El proveedor dispone de la siguiente API:

Obtener todas las ofertas para un sku específico.**Request Example:**

```
curl --location --request GET '{provider_host}/getAllSkuOffers/:sku'
```

Response Status:

200 OK Successful response

Successful Response Body:

```
{
  "sku": "xxx", //string
  "offers": [
    {
      "id": 0, // integer
      "price": 00.00, // decimal
      "stock": 0, // integer - Cantidad de ofertas disponible
      "shipping_price": 0.00, // decimal
      "delivery_date": 2023-05-27, // date
      "can_be_refunded": true, // boolean - Determina si un producto
      tiene devolución.
      "status": "new", // string (new,used,renew)
      "guarantee": true, // boolean - Determina si un producto tiene
      garantía.
      "seller": {
        "name": "xxxx", // string
        "qualification": 0, // integer - Range: 0-5 - Promedio de
        calificaciones.
        "reviews_quantity": 0, // integer - Cantidad de
        calificaciones tiene el seller.
      }
    },
  ],
}
```

```
...  
]  
}
```

Error Response Body:

"error": "Reason"//string -> Descripción del error

Requerimientos:

1. Disponer un servicio que consuma el servicio de nuestro proveedor y determine en base a un criterio abierto que debes definir como la mejor oferta retornando la informacion de la misma.
 2. Disponer un servicio que registre las ventas que se realizan de cada sku y oferta, esta info debe generarse individualmente por orden para poder ser consultada por el equipo de datos.
 3. Debe generarse un reporte diario en una tabla secundaria que resuma las ventas del día por sku.
- No hay limitaciones con respecto a lenguaje o plataforma donde se requiera desarrollar. Se valorará el optar por implementar esto sobre Magento 2.

Nice to Have

- Reflejar el precio de la mejor oferta en el producto.
- Al venderse el producto en Magento registrarse la venta desde aquí.