

# Aplicación Control de Stock

## Objetivo

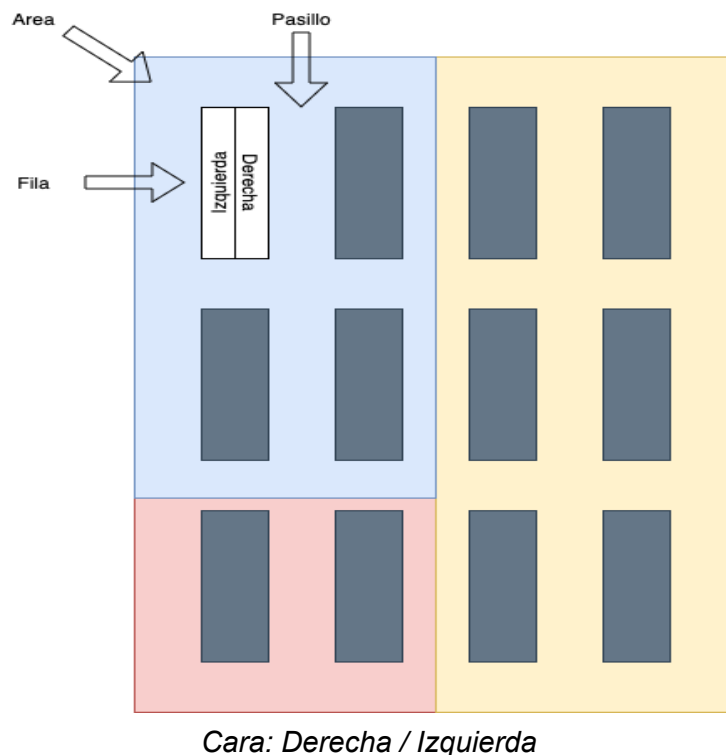
El Negocio de MercadoLibre se sigue expandiendo y ahora decidió incursionar en el almacenamiento de los productos de los vendedores.

Para ese fin, nos fue requerido construir un microservicio para mantener un control de stock de los productos.

Esta aplicación debe mantener la relación entre los productos publicados en el marketplace y su ubicación dentro de cada depósito.

## Consideraciones

Considere el depósito lleno de góndolas como la de los supermercados, y este a su vez dividido en áreas de la siguiente forma.



Por lo que las ubicaciones se definen con el siguiente patrón {Area}-{Pasillo}-{Fila}-{Cara} con 2 dígitos para cada parte.

Siendo el área celeste Limpieza, se identifica la primera ubicación (arriba a la izquierda) como LM-00-00-IZ. O considerando la amarilla como Almacén la última ubicación (abajo a la derecha) se identifica como AL-04-02-DE.



Para el ejercicio no validamos la existencia de las ubicaciones, solo que cumpla con el patrón ({Area}-{Pasillo}-{Fila}-{Cara}).

Los productos publicados en el marketplace pueden estar o no almacenados en nuestros depósitos y podemos validarlo consultado una API pública ([documentación](#)). Si el atributo **shipping.logistic\_type** es **fulfillment**, quiere decir que está ubicado dentro de nuestros depósitos.

Por ejemplo para el ítem [MLA813727183](#). Si consultamos la api de ítems obtenemos.

```
api.mercadolibre.com/items/MLA813727183
{
  "thumbnail_id": "952907-MLA40363363892_012020",
  "thumbnail": "http://http2.mlstatic.com/D_952907-MLA40363363892_012020-I.jpg",
  "secure_thumbnail": "https://http2.mlstatic.com/D_952907-MLA40363363892_012020-I.jpg",
  "pictures": [...],
  "video_id": null,
  "descriptions": [...],
  "accepts_mercadopago": true,
  "non_mercado_pago_payment_methods": [
  ],
  "shipping": - {
    "mode": "me2",
    "free_methods": [...],
    "tags": [
    ],
    "dimensions": null,
    "local_pick_up": false,
    "free_shipping": true,
    "logistic_type": "fulfillment",
    "store_pick_up": false
  },
}
```

Por último, Mercadolibre cuenta con varios depósitos por país, los cuales se identifican con un Id de 4 caracteres, los 2 primeros son el código ISO de país y los otros dos son una secuencia.

Por ejemplo AR01 para el primer abierto en Argentina o BR05 para el quinto abierto en Brasil.

## Requerimientos

1. Exponer un endpoint REST para agregar productos en una ubicación.
  - a. Se nos indicará el Depósito, producto, cantidad y ubicación donde quiere colocar.
  - b. Validar que la dirección tenga el patrón correcto.
  - c. Que el producto/item sea almacenado en nuestros depósitos.
  - d. No se pueden colocar más de 3 productos distintos en una ubicación.
  - e. La suma de las cantidades de los productos que hubiera en una ubicación no puede ser mayor a 100 unidades.
2. Exponer un endpoint para poder retirar productos de una ubicación.  
Se nos indicará el depósito, producto, cantidad y ubicación de donde sacarla.
3. Exponer un endpoint de lectura. Se nos indica un depósito y una ubicación, y este liste los productos y cantidad que hay en el mismo.
4. Exponer un endpoint de búsqueda. Se nos indica el depósito y producto, y este nos devuelva las posibles ubicaciones y cantidad en las mismas.

## Entregable

Para todos los endpoint definir la URL de servicio y el json tanto de pedido como la respuesta.

Se debe utilizar algún motor de base de datos, sea relacional o no.

Indicar consideración y validaciones adicionales que tuvo en cuenta.

Implementar en Java con Spring Boot.

Se entrega mediante un repositorio en Github con el código, scripts de base de datos y documentación que considere necesaria ahí mismo.