Arquitectura:

ServidorLocal

<<servicio>>

Api ML

<<Repositorio>>

Archivos

<<BD>>

SQL

<<WebApp>>

Trazabilidad de Piezas

Servidor MercadoLibre

<<Aplicación>>

Geminis

Servidor VanGogh

Persistencia:

Pieza

Nombre: Varchar

Tipo: Varchar

IdRegion: Varchar

MotivoBaja: Varchar

IdPieza: Varchar PK

Estado: Integer

idPersona: Integer

ValorCompra: Integer

FechaCompra: DateTime

Region

Ciudad: Varchar

Estado: Varchar  
Pais: Varchar

IdRegion: Varchar PK

Persona

IdPersona: Integer PK

Telefono: Integer

PersonaFisica

Nombre: Varchar

Apellido: Varchar

DNI: Integer

IdPersona: Integer FK

PersonaJuridica

Razon: Varchar

CUIL: Integer

IdPersona: Integer FK

Movimientos

IdOrigen: Varchar

IdDestino: Varchar

Fecha: DateTime

IdMovimiento PK

IdPieza

Sala

IdSala: Varchar PK

Descripcion: Varchar

Reportes

IdReporte: Integer PK

Tipo: Varchar

Fecha: DateTime

Acciones

IdAccion: Integer PK

IdPieza: Varchar

Descripcion: Varchar

Justificaciones:

La herencia de Persona, PersonaFisica y PersonaJuridica la refleje como 3 tablas en el modelo de datos, esto fue porque todos los atributos de ambos objetos heredados eran diferentes. Usando esta solución, el ID de cada PersonaJuridica y PersonaFisica es refenciado desde el de Persona que es único para cada Persona.

Por otro lado, la herencia de Pieza, PiezaDonada y PiezaAdquirida la refleje como 1 sola tabla, por el hecho de que la mayoría de los atributos coincidían y solo se le agrega los campos de Compra que para el caso de PiezaDonada pueden ir en 0 o Null.

Los patrones como el State y Strategy y la clase de generación de reportes, no fueron mapeadas al DER porque no persisten nada.

Los ID de Pieza y Sala elegí que sean Varchar para poder usar mejor el sistema de barras. Los demás son Autoincrementados (excepto el de región que se usa el provisto por la API de ML).  
Los formatos de Fecha fueron DateTime para poder manejarlos mejor desde el código. Igualmente dependiendo del motor de BD podría usarse como varchar y luego serializarse y deserializarse cuando sea pasado entre el programa y la BD.

La API de ML facilita la generación de las regiones, pero por las dudas se tienen persistidas enteras (no solo el código de región) por dos razones. La primera es que la conexión a la API tarda mucho y la segunda es que, si la API llegara a dejar de funcionar, se perderían todas las regiones quedándonos solo con los códigos.