	INTRASOFT		Vigente a partir de: 2018-03-16
	Prueba Técnica – Desarrollo		Versión 1.0
			Página 1 de 2

Prueba Técnica: API Activos Fijos.

Los activos fijos de una empresa son sus bienes tangibles, son necesarios para el funcionamiento de la mayoría de empresas y no se destinan para la venta. Ejemplos de estos son: bienes inmuebles, maquinaria, material de oficina, etc.

Estos activos fijos tienen características comunes entre todas como son: nombre, descripción, tipo, serial, numero interno de inventario, peso, alto, ancho, largo, valor compra, fecha de compra, fecha de baja, estado actual (activo, dado de baja, en reparación, disponible, asignado), color.

Los activos se pueden asignar a un área o persona, si es a un área esta pertenece a una ciudad (diferentes áreas se encuentran en diferentes ciudades).

Diseñar una API RESTFul para administrar los activos, esta debería permitir:

- Buscar por todos los activos.
- Crear nuevos activos.
- Actualizar activos.
 - o Permite cambiar el serial interno y fecha de baja.

Se requiere realizar el diseño del entregable, se solicitan los siguientes artefactos:

- **Obligatorios:**
 - Modelo de datos
 - Diagrama de casos de uso
 - Pruebas Unitarias (JUnit).
 - Mockup.
- **Opcionales**
 - Diagrama de secuencia para un caso de uso
 - Diagrama de estados (si aplica)
 - Diagrama de arquitectura de referencia

Plataforma:


Es libre de seleccionar como base de su solución alguna de las siguientes tecnologías:

- Spring.
- Spring + JEE.

Tecnologías Extras

Debe utilizar alguna de las siguientes tecnologías.

- JAVA 7 o superior.
- Spring

	INTRASOFT		Vigente a partir de: 2018-03-16
	Prueba Técnica – Desarrollo		Versión 1.0
			Página 2 de 2

Base de datos:

Es libre de utilizar cualquier motor de base de datos **NoSQL**, pero se recomienda el uso de MongoDB. Se requiere justificar la selección de motor y ventajas.

Reglas

- El código debe estar versionado en GIT.
- Se deben validar las fechas (nunca una fecha de baja debe ser superior a una fecha de compra).
- No debe haber excepciones sin capturar.
- Las repuestas de la API deben estar estandarizadas a:
 - 200 Por consultas exitosas.
 - 400 Para reportar datos faltantes al crear o actualizar un activo.
 - 404 para búsquedas sin resultados.
 - 500 Para errores que pasen en la capa de backend.
 - Todas las respuestas deben contener una descripción o resumen de lo que paso.

Entregables:

- Repositorio con el código donde se puedan ver los commit realizados.
- WAR, EAR, JAR, con las instrucciones de generación.
- Instrucciones de despliegue.
- Modelo Entidad Relación Base De Datos
- Archivo de pruebas en POSTMAN.

PLUS

Los siguientes puntos no son requeridos en su solución, pero si los decide incluir serán tenidos en cuenta.

- Documentación de la API (RAML, Swagger).
- Documentación Java (JAVADOC).
- Repuestas múltiple de la API (JSON, XML).
- Log (log4j).
- Cliente javascript (Angular, React, etc), que consuma algún servicio.
- Definición de requerimientos funcionales y no funcionales