

# Sistema estacionamiento inteligente (Ejercicio Auto estudio)

---

Los centros comerciales de la ciudad desean hacer piloto para estudiar el desarrollo de un sistema de IoT para el estacionamiento inteligente.

Este sistema se compone de:

## 1. Hardware

Se desea instalar un nodo de IoT con sensor magnético (magnetómetro) que detecta los autos desde el piso de cada celda de estacionamiento. Cada minuto estos nodos se comunican a la central y transmiten su identificación y si está libre u ocupado (0 o 1) [**Por ahora NO hay más estados posibles**].

Estos identificadores de nodos están asociados a un **centro comercial** y una **ubicación geográfica** dentro de cada **zona de parqueaderos** y a uno o más **entradas vehiculares** (que permiten el fácil acceso a esta zona).



## 2. Software

Se desea desarrollar dos módulos:

*2.1- APP que le permite a los usuarios consultar en tiempo real el estado del estacionamiento, así:*

- Lista los centros comerciales

## Examen de técnicas avanzadas de base de datos

### Diseño base de datos NoSQL

- Selecciona el Centro comercial, La aplicación le muestra el número de celdas libres por cada zona de parqueaderos. Nota 1 y 2



- Luego El usuario selecciona una zona de parqueaderos y el sistema le muestra todos los sensores y su estado (libre u ocupado). Nota 1 y 2



- Finalmente le lista las ubicaciones de las Entradas Vehiculares si esta interesada en usar la zona de parqueaderos.

## 2.2- Aplicación WEB con las estadísticas de uso de parqueaderos

2.2.1 Se selecciona una fecha y el Centro comercial. Y la aplicación le muestra el porcentaje promedio de uso por día (Sumatoria de minutos que estuvo ocupado cada celda dividido el número total de celdas por 840 [que equivalen a 14 horas en minutos]). Nota 1

### Notas:

1. Para hacer el pantallazo es muy probable que requiera más de una consulta.

## Examen de técnicas avanzadas de base de datos

### Diseño base de datos NoSQL

2. Para Consultar el ultimo estado de las celdas en cada zona. Simplemente haga que se ordene por hora (tiempo) de forma descendente y se toma el primer registro recuperado.

### Entregas

1. Imagen del diagrama E-R
2. Hacer diagrama de WorkFlow y Diagramas de chebotko (Given y Find)
3. Archivo texto plano de la creación de las tablas y el keyspace (.cql)
4. Archivo texto plano de la inserción de datos de ejemplo de las tablas (.cql)
5. Hacer ejemplo de la ejecución de las consultas del WorkFlow (Puede detallar en el prototipo presentado cuales consultas se usan para cada pantallazo) (.cql)
6. Opcional: Archivo de texto con ejemplos de las colecciones para Mongo. Asuma que hay al menos dos centros comerciales, que cada centro comercial tiene al menos 2 zonas y cada zona tiene 3 celdas.