

Arquitectura de Datos



CLASE 3 - 2:13:38 HABLA DE LAS QUERIES DEL TP

Problemática a resolver

Se pretende realizar un sistema de *geolocalización* mediante pequeños dispositivos diseñados para tal fin, los cuales llamaremos *dispositivos de seguimiento* de aquí en adelante. Dicho sistema será aplicado al seguimiento de vehículos.

Descripción de la aplicación

El seguimiento sería realizado mediante una aplicación en la cual cada **usuario**:

1. visualice y gestione sus **dispositivos**,
2. gestione la información por estos brindada,
3. configure **alarmas** de interés.

Alcance

En principio, la aplicación se limita a conocer las coordenadas GPS de los dispositivos de seguimiento en intervalos fijos de tiempo.

La información fundamental que dichos dispositivos deben transmitir luego de cada muestreo es:

1. Código identificador del dispositivo (*Device_ID*)
2. Coordenadas GPS
3. Fecha y hora (*TimeStamp*) a las cuales fueron tomadas las coordenadas.

Métricas útiles

Las métricas dependerán de la aplicación que se quiera dar al sistema de seguimiento. Sin embargo, en su mayoría, las métricas que más se utilizan en este tipo de sistemas contribuyen a reconstruir caminos o conjuntos de coordenadas en determinados intervalos de tiempo.

Ejemplos:

- **Seguimiento de vehículos de transporte:** conjunto de coordenadas GPS donde se ubicó un determinado vehículo entre la hora de salida y de llegada.
- **Geoperimetraje:** fechas y horas durante las cuales un determinado dispositivo estuvo fuera del área permitida.

Requerimiento:

- Proponer el DER necesario para esta solución
- El modelo debería ser capaz de resolver las siguientes consultas:

- Última información de los vehículos dentro de un determinado área, tal que la antigüedad del dato de posición no sea superior a una dada cantidad de tiempo.
- Posiciones conocidas de un dado vehículo en un dado período de tiempo.

Anexo I: DDL de la base de datos relacional propuesta

```
CREATE DATABASE tp_tracking_relacional;
```

```
USE tp_tracking_relacional;
```

```
CREATE TABLE usuario (  
    user_id INT NOT NULL AUTO_INCREMENT,  
    dni INT(9),  
    apellido VARCHAR(20),  
    nombres VARCHAR(60),  
    direccion VARCHAR(60),  
    PRIMARY KEY (user_id)  
);
```

```
CREATE TABLE tracking_device (  
    device_id INT NOT NULL,  
    PRIMARY KEY (device_id)  
);
```

```
CREATE TABLE medicion (  
    medicion_id INT NOT NULL AUTO_INCREMENT,  
    time_stamp TIMESTAMP,  
    lat FLOAT(10 , 6 ),  
    lng FLOAT(10 , 6 ),  
    PRIMARY KEY (medicion_id)  
);
```

```
CREATE TABLE usuario_tracking_device (  
    user_id INT NOT NULL AUTO_INCREMENT,  
    device_id INT NOT NULL,  
    PRIMARY KEY (user_id , device_id),  
    FOREIGN KEY (user_id)  
        REFERENCES usuario (user_id),  
    FOREIGN KEY (device_id)  
        REFERENCES tracking_device (device_id)  
);
```

```
CREATE TABLE tracking_device_medicion (  
    device_id INT,  
    medicion_id INT,  
    PRIMARY KEY (device_id , medicion_id),  
    FOREIGN KEY (device_id)  
        REFERENCES tracking_device (device_id),  
    FOREIGN KEY (medicion_id)  
        REFERENCES medicion (medicion_id)  
);
```

Anexo II: diagrama UML

A continuación se muestra el diagrama UML de la base de datos relacional propuesta, obtenido mediante *reverse engineering* en MySQL Workbench.

