

Entregables: Código fuente, tests y un pequeño archivo de texto explicando los aspectos más relevantes de la implementación así como las decisiones tomadas.

Python

1.- Implementar un endpoint de una api REST en python que reciba peticiones con los siguientes parámetros:

Nombre de usuario

Código Postal

2.- A partir del CP, y utilizando el servicio de geonames, obtenemos la ciudad del usuario (<http://www.geonames.org>)

3.- Almacenamos la información en bdd, en dos tablas. En la tabla master guardamos el nombre del usuario. En la tabla detalle guardamos su cp y su ciudad.

4.- El servidor responde en formato JSON, indicando si todo ha ido bien o si ha habido algún tipo de error. Se debe pensar que este endpoint formará parte de un conjunto de endpoints distintos que no se deben programar para este ejercicio, pero se debe tener en cuenta a la hora de crear la estructura de software que permita extender/añadir nuevas funcionalidades de forma simple.

Opcional:

5.- Utilizando TypeScript (nodejs) crea un endpoint para listar qué usuarios residen en un código postal concreto.

6.- Utilizando TypeScript (nodejs) crea un endpoint para eliminar todos los usuarios que residen en un código postal concreto.

Opcional:

7.- Utilizando Java crea un endpoint para devolver la estadística de cuántos usuarios viven en cada código postal

8.- Utilizando Java crea un endpoint que retorne: Cuál es el CP con más usuarios (y cuántos usuarios son), cuál es el CP con menos usuarios (y cuántos usuarios son), cuál es la media de usuarios por CP y cuál es la desviación típica.

Entregables: Código fuente, tests y un pequeño archivo de texto explicando los aspectos más relevantes de la implementación así como las decisiones tomadas.