1. Generar un proyecto con las siguientes características

   A. El proyecto debe exponer los siguientes servicios REST (Persistencia de datos se debe realizar utilizando un esquema local en MySQL)
      a. Servicio de creación de usuarios que deberá recibir
         i. Name
         ii. Username
         iii. Email
         iv. Phone
      b. Listado de usuarios registrados
      c. Obtención de usuario registrado mediante email
      d. Eliminar un usuario
      e. Los servicios de **creación y eliminación deben estar protegidos** con seguridad (como opción puede ser Spring Security, utilizando un usuario y contraseña que esté dentro de los properties de la aplicación)
   B. Generar un servicio POST que consultará a una API externa y que deberá considerar la recepción de un parámetro (param)
      a. Servicio deberá cifrar el parámetro utilizando DES con la llave "ionix123456"
      b. Se deberá invocar la siguiente URL entregando el parámetro cifrado como valor (se debe incluir el header indicado para poder realizar el consumo)
         i. GET https://my.api.mockaroo.com/test-tecnico/search/{parametro_cifrado}
         ii. {parametro_cifrado} es el valor que se recibe en el servicio (param) cifrado con DES
         iii. Header- > X-API-Key: f2f719e0
      c. El valor parámetro de prueba es "1-9"
   C. Servicio debe retornar un objeto JSON en el formato indicado con la cantidad de registros devuelto por el servicio llamado anteriormente. En el campo "elapsedTime" se debe entregar el tiempo de ejecución del servicio "search" en milisegundos

```
{
        responseCode: 0
        description: "OK",
        elapsedTime: 245,
        result: {
                registerCount: NN
        }
}
```

Ejemplo de implementación de cifrado en DES

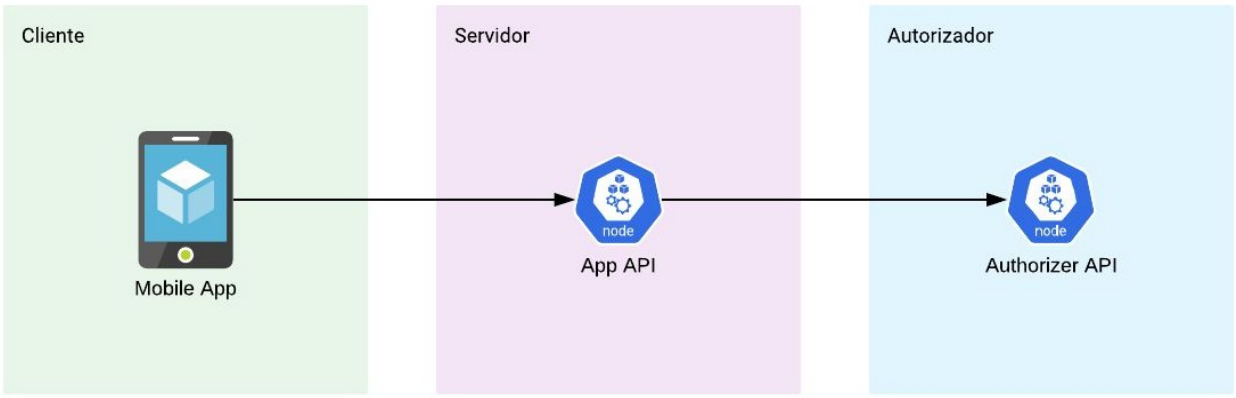
```
DESKeySpec keySpec = new DESKeySpec("ionix123456".getBytes("UTF8"));
SecretKeyFactory keyFactory = SecretKeyFactory.getInstance("DES");
byte[] cleartext = plainRut.getBytes("UTF8");
Cipher cipher = Cipher.getInstance("DES");
String encryptedRut = Base64.encodeToString(cipher.doFinal(cleartext), Base64.DEFAULT);
```

La implementación del ejercicio debe considerar cobertura de test unitarios, documentación, readme o cualquier adicional que evalúe como buena práctica.

Ejercicio 2

Considerar el siguiente caso:
Una aplicación debe capturar data sensible y enviarla a un servicio de autorización. La data sensible debe ser transmitida a través de la APP API según se indica en el siguiente diagrama, sin embargo, no puede ser legible hasta que llegue al servicio de autorización.

Describa un método para permitir la captura y transmisión de información de forma segura según lo planteado, sin considerar el uso de certificados SSL.



Importante: El desarrollo de este ejercicio no considera una implementación, si no que la generación de una propuesta de solución para el escenario planteado.