Crear un ServicioWeb en Java que reciba 2 parámetros (Numero documento, Nombre), valide que ambos campos no sean nulos y retorne un campo con el mensaje "¡DATOS RECIBIDOS!", en caso de que la validación falle retorne "¡DATOS INCORRECTOS!", se debe imprimir en un log el request y el response de la transacción.

OPCIONAL: desencriptar el contenido del campo Numero documento, utilizando la clave privada enviada a continuación e imprimirla en el Log.

“-----BEGIN RSA PRIVATE KEY-----

MIICWwIBAAKBgFziZlz1VnGhqISYCoKet7ED8pEJU5Y3XoZ7Pep8LCFUlKZ+bZTm

gq4gawbpVtUCMJlTIDyQcT2zlzyBDLbBPgsDdEw868F9TioOjbQ+l6dfrXIuaRR3

n8+IKEx2NIP0HtwtIjwKNv1nhbmEttYau/fAtxi/Xvw2mmAXi+e3kFJPAgMBAAEC

gYARcIzJrWLSqQisBKQMjGJvAQ+9PaQNE05TL7abXT8n7uqOLDTE/VbR/NI7lPox

ulyHnTzzQieQ7zRZLt3FPpk4W612W35aoiusLMY9Tl9+IJTJH/fJWoVTOT6jpllj

LL87t1R0gbl6a1/8n4hp1X6+qAJxlG3KvEKNU3bxhUhamQJBALGG3qpOTyvS0vp+

pkyxPjvV8VwplMu27klS469uByPpIVJ6UVYGbgWhGzUsaPVKmEIEoEbk9mj2VfL3

RVBfIDsCQQCF8UwEEXiMeiWogu8kNapnAC6l/6SSiOnuDWrWG4aRXyRiRviTptS6

Hd9gZENfZG3wObrBrh0lGN2kQKWPiOj9AkEArshmua5X7IGpDs9a0+89opPFCkQ2

J0t31+EwIixmA0kocZfUNKon8IrpyrRqsfY7aeQ8GRCcOkMt5ATnzXWauQJAHvcU

2s/rLZbDg/yZKqbZeSx6nFfIhTPv2N/zNgJxDsSPFcVQjFoCTfDABnnHdZMzM1k1

Srdk94GTI/jqDY/aRQJASk5i1+D7TWL0YcwWE1wi+rMC1YgBdqJYFjKQF5sy0IbZ

mkCSJbrZ2RpFY2voX9ZBip8c1O7A4oXn8RvvRbimAA==

-----END RSA PRIVATE KEY-----”

Debes crear una aplicación Angular5 con un formulario que tenga los siguientes campos de entrada:

1. Número de documento

2. Nombre

3. Botón para enviar el formulario.

4. Un campo oculto llamado documentoCifrado

Al cargar o iniciar la aplicación debes llamar un API que te retorne una clave publica en formato PEM (la cual adjunto más abajo), esta debe quedar en el contexto de aplicación Angular como una variable global.

Cuando el usuario ingrese el número de documento y pase al campo de nombre, en el campo número de documento debe quedar el valor enmascarado con los últimos 2 dígitos visibles, es decir, si el usuario ingresó 1234567, se debe ver \*\*\*\*\*67. Y en el campo oculto debe quedar el número de documento cifrado con la llave pública. (Esto lo hace la función indicada más abajo).

Al enviar el formulario, se debe invocar un API que reciba el número de documento (encriptado) y el nombre, luego consuma el webservice implementado en el 1er punto y retorne la respuesta del webservice, el cual se debe mostrar en pantalla.

Las APIs deben quedar implementadas de forma que, si invocas el API por el browser, esta debe retornar la data en formato JSON.

Debes usar la siguiente clave pública, (el contenido entre comillas):

"-----BEGIN PUBLIC KEY-----

MIGeMA0GCSqGSIb3DQEBAQUAA4GMADCBiAKBgFziZlz1VnGhqISYCoKet7ED8pEJ

U5Y3XoZ7Pep8LCFUlKZ+bZTmgq4gawbpVtUCMJlTIDyQcT2zlzyBDLbBPgsDdEw8

68F9TioOjbQ+l6dfrXIuaRR3n8+IKEx2NIP0HtwtIjwKNv1nhbmEttYau/fAtxi/

Xvw2mmAXi+e3kFJPAgMBAAE=

-----END PUBLIC KEY-----“

Debes utilizar el archivo (cambiar extensión .txt a .js). Este archivo tiene una función llamada encriptarCampo, la cual recibe 3 parámetros:

* nomVar: El identificador del campo del número de documento
* nomVarCrypto: El identificador del campo oculto que almacenará el dato cifrado
* visibles: La cantidad de dígitos visibles en el campo enmascarado (en nuestro caso debe recibir el valor 2).

Esta función enmascara el dato ingresado en el primer parámetro, y el resultado queda en el mismo campo, también cifra el dato con la llave publica, la cual debe estar almacenada en una variable global llamada publicKey, y el resultado lo almacena en el campo oculto.

En resumen, cuando el usuario salga del campo número de documento, se debe invocar la función encriptarCampo para que enmascare y cifre el dato en el formulario. Al enviar el formulario al API, si revisamos las peticiones por las herramientas de desarrollo del browser, el número de documento no debe viajar en claro, sino cifrado.

El objetivo principal, es cifrar el número de documento con la librería entregada y enviarlo al API sin que se vea el texto en claro. Si para cumplir el objetivo requieres hacer cambios en las funciones, solo puedes modificar las dos últimas funciones de la librería, las demás deben permanecer iguales.

Los Entregables son:

* Fuentes del desarrollo.
* Logs de las transacciones.
* Capturas de pantallas de la prueba de la transacción.