

Evaluación Plataformas Especiales

Se debe realizar la siguiente aplicación basada en API's para procesar los datos de una transacción como se muestra a continuación:

La primer API consume el siguiente JSON

```
{
  "operacion": "venta",
  "importe": "100.00",
  "cliente": "Carlos",
  "firma": "jejdjw134&3#$$"
}
```

La cual deberá de cumplir con los siguientes puntos:

-Validar los atributos del JSON utilizando alguno(s) de los siguientes métodos:

-@Valid, @Regex, @RestControllerAdvice, para el atributo operacion se validará que sean caracteres, para el atributo importe se valida que sea de tipo moneda y para cliente que sean caracteres, la longitud queda a criterio del desarrollador.

-El atributo firma se calcula con el algoritmo SHA (enviado desde el front), por lo que al llegar al API, se volverá a calcular con los atributos "operación", "importe" y "cliente" y se compara con el de la petición JSON, si son iguales continuara al siguiente paso, en caso contrario se retorna bad request .

-Si el formato del JSON es válido se deberá de mandar por medio de @OpenFeign, RestTemplate o Apache Client a la segunda API La segunda API deberá de cumplir con lo siguiente:

-Se almacenarán los datos recibidos utilizando SpringData mediante JpaRepository.

La estructura de la base de datos es la siguiente:

PK, operacion, importe, cliente, referencia, estatus, secreto La base de datos sera H2 (base de datos en memoria se configura en el pom.xml y application properties.

Una vez guardada la información se generará una referencia numérica aleatoria de 6 dígitos y en el estatus indicaremos la leyenda Aprobada, se le informará al primer API esta información para generar un response como se muestra a continuación:

```
{
  "id": "2376",
  "estatus": "Aprobada",
  "referencia": "262737",
}
```

```
        "operacion": "venta"
    }
}
```

*El atributo id es la PK del registro almacenado.

Para poder enviar las peticiones al API se deberá de generar un front (página web) utilizando JavaScript, se puede utilizar Angular, Vue.js, React.

Ejemplo del front para login y registrar la información:

Ventana de Login

Textfield usuario: _____

Textfield password: _____

Para validar el password, creamos una tabla en base de datos en la cual tendremos una columna de usuario y password, este password se tiene que cifrar con el algoritmo Bcrypt (previamente, antes de iniciar sesión, podemos insertar esta información a la base con un script o con un query) para que al momento de iniciar sesión compare el usuario y password y de ser correctos pasemos a la siguiente ventana.

Ventana Registrar operación :

Textfield operación: _____

Textfield importe: _____

Textfield cliente: _____

Textfield firma: _____

Button (Manda petición Post) Al recibir la respuesta del API se mostrará está en una notificación.

*De lado del front se calcula la firma para que el API valide la integridad de los datos (se calcula con el algoritmo sha, utilizando los campos operación, importe y cliente.

*Se utilizará el método Patch para actualizar el estatus de una transacción "aprobada" a "cancelada", para esto solo se mandarán los campos id, referencia y estatus con el valor "cancelar" (Se puede mostrar desde Postman), se debe utilizar @Query

*Se obtendrán todos los registros almacenados por medio de un método Get (Se puede mostrar desde Postman) *Se realizara una consulta utilizando @Query para obtener el registro en base a la referencia