Tecnova
SOLUCIONES INFORMATICAS

Especificaciones actualización validaciones de esquemas

Índice

1. Introducción

El presente documento está dirigido a usuarios de entidad que utilicen o se integren al sistema PIAS para detallar las especificaciones liberadas en Julio del 2026 correspondientes a la actualización de validaciones de los esquemas de autenticación y reconocimiento.
Estos cambios abarcan los siguientes esquemas:

- SCHEMA GENERAL
- SCHEMA AUTENTICACIÓN

Y las siguientes reglas de validación:

- REGEX_PASSWORD
- REGEX_USERNAME
- BLACKLIST_WORDS

Esta actualización afecta los siguientes módulos del sistema:

- Login web
- Cambio de clave Login
- Cambio de clave Menú PIAS
- Creación de usuario (solo campo username)
- Servicio SOAP y REST de autenticación
- Servicio SOAP y REST de ingreso de reconocimiento
- Servicio SOAP y REST de consulta de reconocimiento
- Servicio SOAP y REST de actualización de reconocimiento
- Servicio SOAP y REST de extinción/anulación de reconocimiento

Eliodoro Yañez 2239, Santiago, Chile.
Telefono: +562 2714 0700
www.tecnova.cl

2.Validaciones en Servicios

Detalle de la estructura de JSON Schema para el uso de servicios REST de reconocimiento, los cuales contienen dos campos: “token” y “xml”. En esta actualización se aplican mejoras en la validación de esta estructura y en sus respuestas.

2.1.JSON SCHEMA para api´s de reconocimiento:

- POST /api/servicio/reconocimiento/service
- POST /api/servicio/reconocimiento/causante/service
- POST /api/servicio/reconocimiento/extincion/service
- POST /api/servicio/reconocimiento/anulacion/service

JSON

```
{
  token: {
    in: ['body'],
    errorMessage: 'El token es requerido.',
    isString: true,
    trim: true,
    notEmpty: true,
  },
  xml: {
    in: ['body'],
    errorMessage: 'El XML es requerido.',
    isString: true,
    trim: true,
    notEmpty: true,
  }
}
```

Este esquema se resume a que para ambos campos: son obligatorios, no se permiten vacíos y no se pueden enviar solo con espacios en blanco. Además, se valida que el valor de ambos campos debe ser un texto, esto último considerando que el campo “xml” es un texto escapado de un xml utilizado para los servicios SOAP.

Para el caso del servicio de autenticación, se detalla a continuación la estructura de su JSON Schema el cual posee los campos: “codigoEntidad”, “loginUsuario” y “claveUsuario”. Además se agregan las reglas de validación: REGEX_PASSWORD, REGEX_USERNAME y BLACKLIST_WORDS.

2.2.JSON SCHEMA para api de autenticación:

- POST /api/servicio/autentificacion/login/service

JSON

```
{
  codigoEntidad: {
    in: ['body'],
    isNumeric: true,
    exists: true,
    errorMessage: 'Código entidad inválido'
  },
  loginUsuario: {
    in: ['body'],
    isString: true,
    trim: true,
    escape: true,
    exists: true,
    isLength: { options: { min: 1, max: 100 }, errorMessage:
'Supera el máximo de caracteres' },
    matches: { options: StringUtils.REGEX_USERNAME },
    custom: {
      options: (value: string) => {
        return StringUtils.blackListWords(value);
      }
    },
    errorMessage: 'Login usuario inválido'
  },
  claveUsuario: {
    in: ['body'],
    isString: true,
    trim: true,
    matches: { options: StringUtils.REGEX_PASSWORD },
    custom: {
      options: (value: string) => {
        return StringUtils.blackListWords(value);
      }
    },
    errorMessage: 'Clave usuario inválido'
  },
}
```

Esto se puede resumir de la siguiente manera:

- El código de entidad es obligatorio, solo recibe números y no permite el envío de solo espacios en blanco.
- El usuario es obligatorio, no permite enviarse vacío o sólo con espacios en blanco y además, tanto el usuario como la contraseña deben cumplir un formato específico de caracteres permitidos (detallado en Descripción de formatos permitidos).
- Además de lo anterior, tanto el usuario como la contraseña son revisados contra una lista de palabras no permitidas; si el valor ingresado coincide con alguna, la solicitud será rechazada.
- El campo de usuario tiene un límite de 100 caracteres.

2.3.Descripción de formatos permitidos.

Campo Nombre de usuario:

- Debe existir en el request body
- Debe ser texto (string)
- Se escapan caracteres especiales (ej: < pasa a ser <)
- Longitud entre 1 y 100 caracteres
- Solo se permiten letras minúsculas (incluida la ñ), números, y los símbolos: . _ @ -
- No se permiten mayúsculas, espacios ni otros símbolos.
- No puede contener palabras de la lista negra: script, from, select, update, delete, insert, drop, alter, table.

Ejemplo de nombre de usuario permitido: **juan.perez_01**

Campo contraseña

- Debe ser texto (string)
- Se recortan espacios al inicio y al final (trim)
- Longitud entre 8 y 16 caracteres
- Al menos una letra minúscula
- Al menos una letra mayúscula
- Al menos un número
- Al menos un carácter especial de los siguientes caracteres: # ? ! = ( ) / . @ $ % , _ ^ & + * -
- No puede contener palabras de la lista negra: script, from, select, update, delete, insert, drop, alter, table

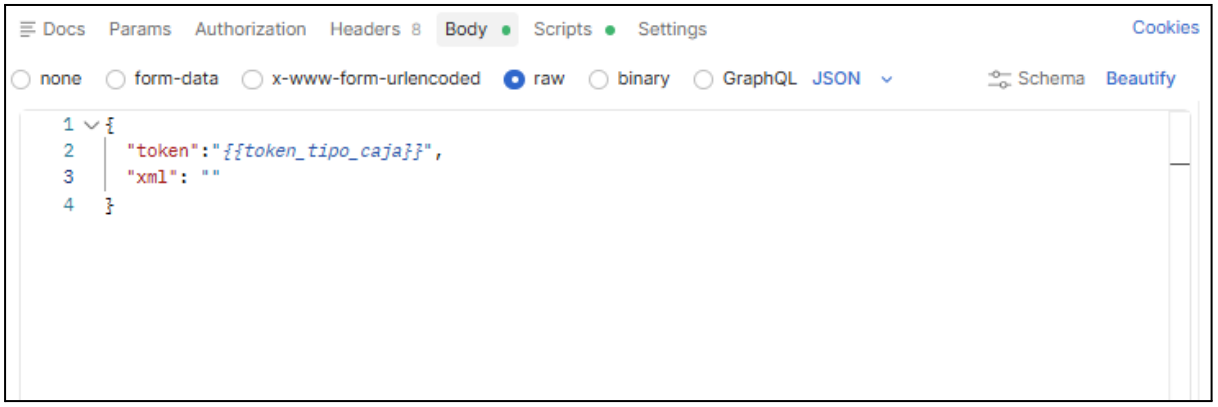
Ejemplo de contraseña permitida: **Clave2026!**

2.4.Ejemplos de funcionamientos:

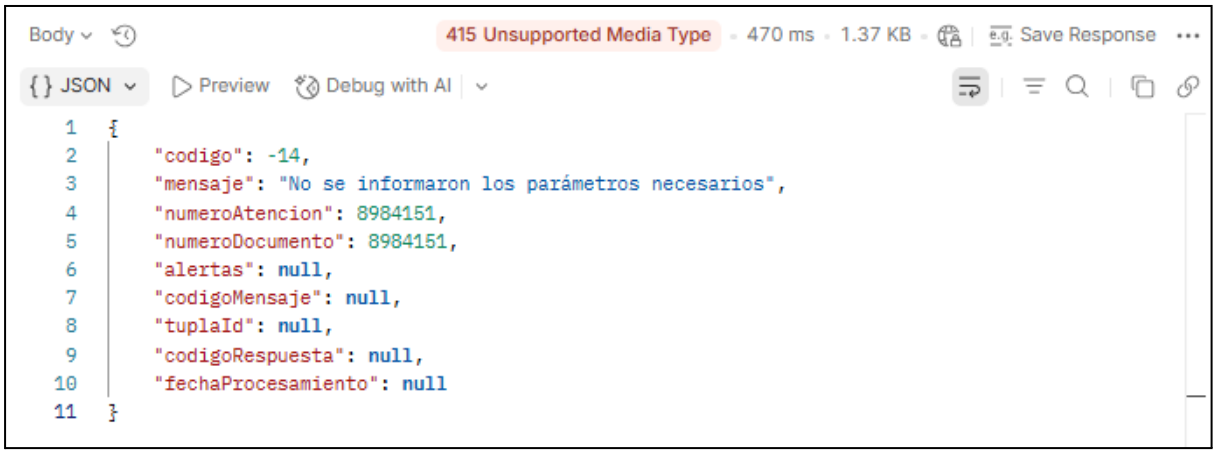
2.4.1. Reconocimiento REST

En el caso de que el el body request de un ingreso de reconocimiento el campo “xml” se encuentre vacío, el sistema responderá con un rechazo en el request:

Body request:



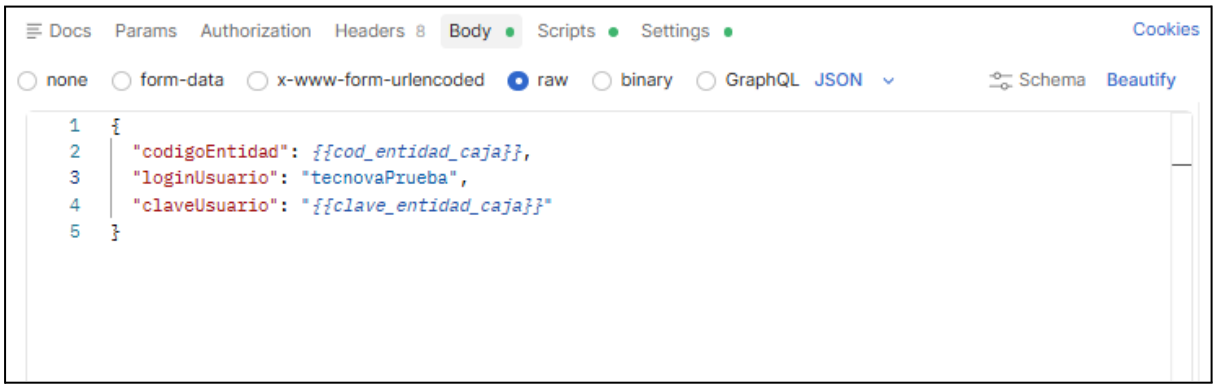
Respuesta servicio:



2.4.2. Autenticación REST, campo “loginUsuario”

Uno de los cambios en el servicio, es que no se permiten letras en mayúsculas para el campo “loginUsuario”, al enviar un request con este tipo de datos será rechazado:

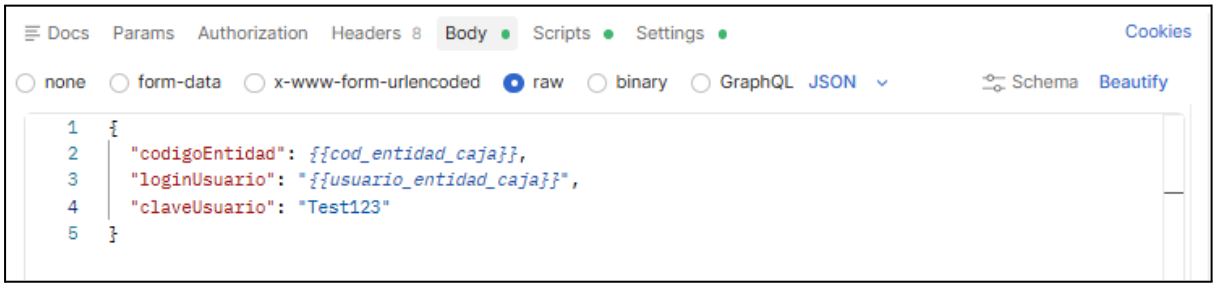
Body request:



Respuesta servicio:

2.4.3. Autenticación REST, campo “claveUsuario”
Esta regla ahora validará que la clave debe contar con mínimo 8 caracteres, por lo que si se envía una contraseña con 7 caracteres o menos se rechazará el request.

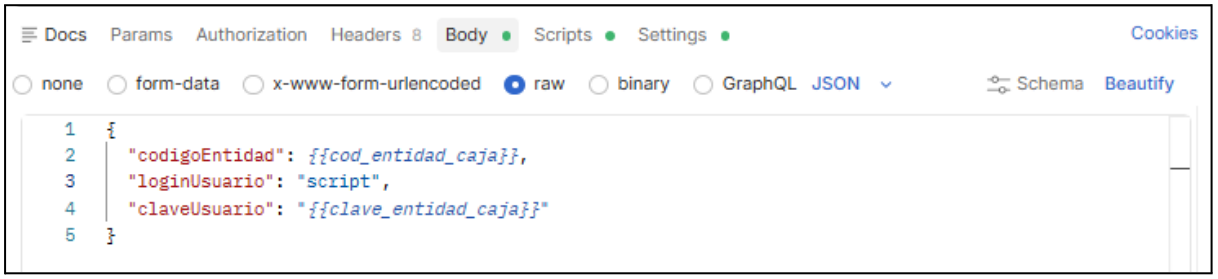
Body request:



Respuesta servicio:

2.4.4. Autenticación REST, palabras prohibidas
Otro ejemplo es el de el uso de palabras prohibidas, para el campo “loginUsuario”:

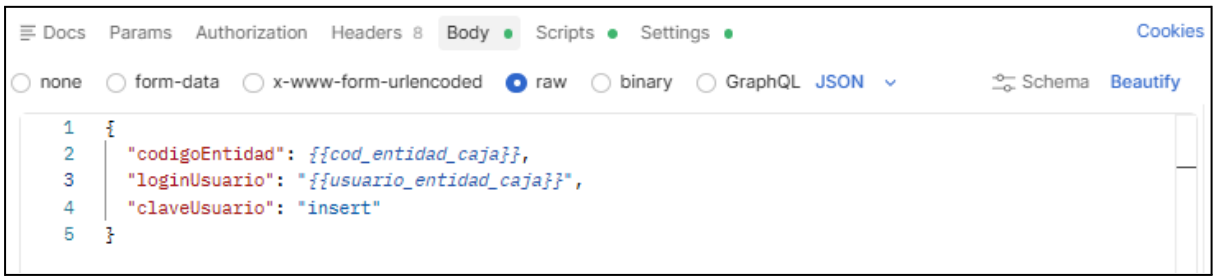
Body request:



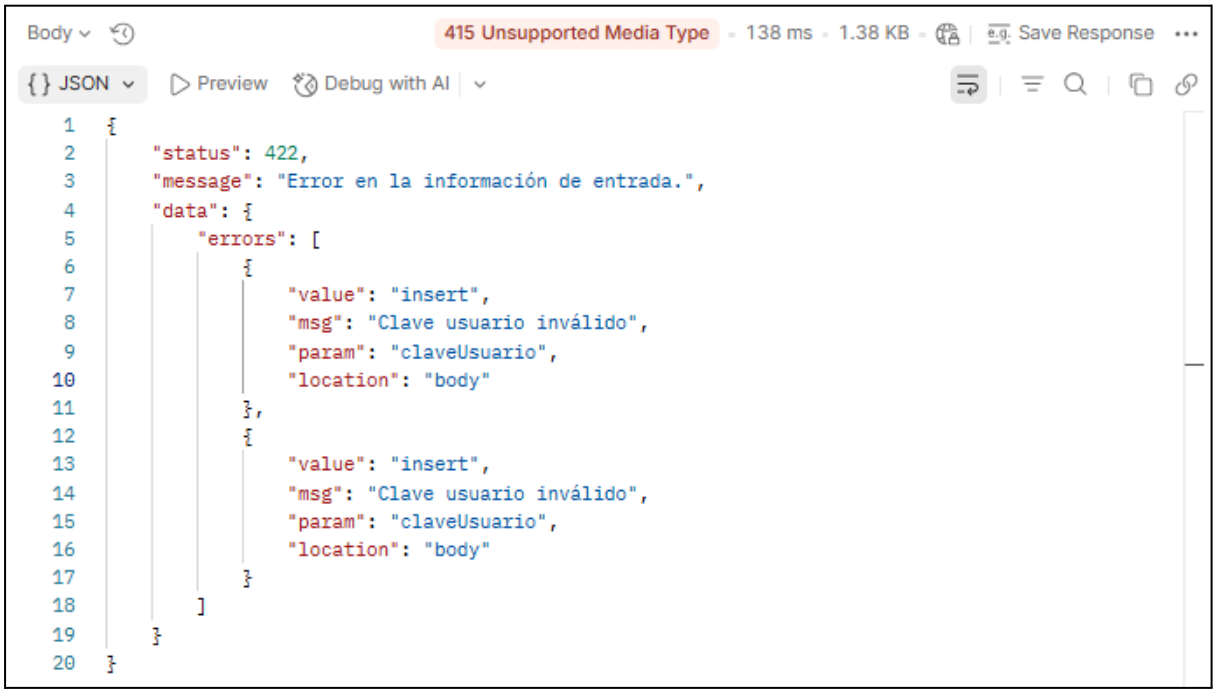
Respuesta servicio:

Y también, para el campo “claveUsuario”:

Body request:



Respuesta servicio:



3.Validaciones en Plataforma Web

Los cambios realizados a los esquemas se aplican de manera transversal al sistema, esto también influye en los módulos web que consumen estos servicios. Módulos web que utilizan estos servicios:

1. Login web
2. Cambio de clave Login
3. Cambio de clave Menú PIAS
4. Creación de usuario (solo campo username)

3.1. Ejemplos de funcionamiento

3.1.1. Login web

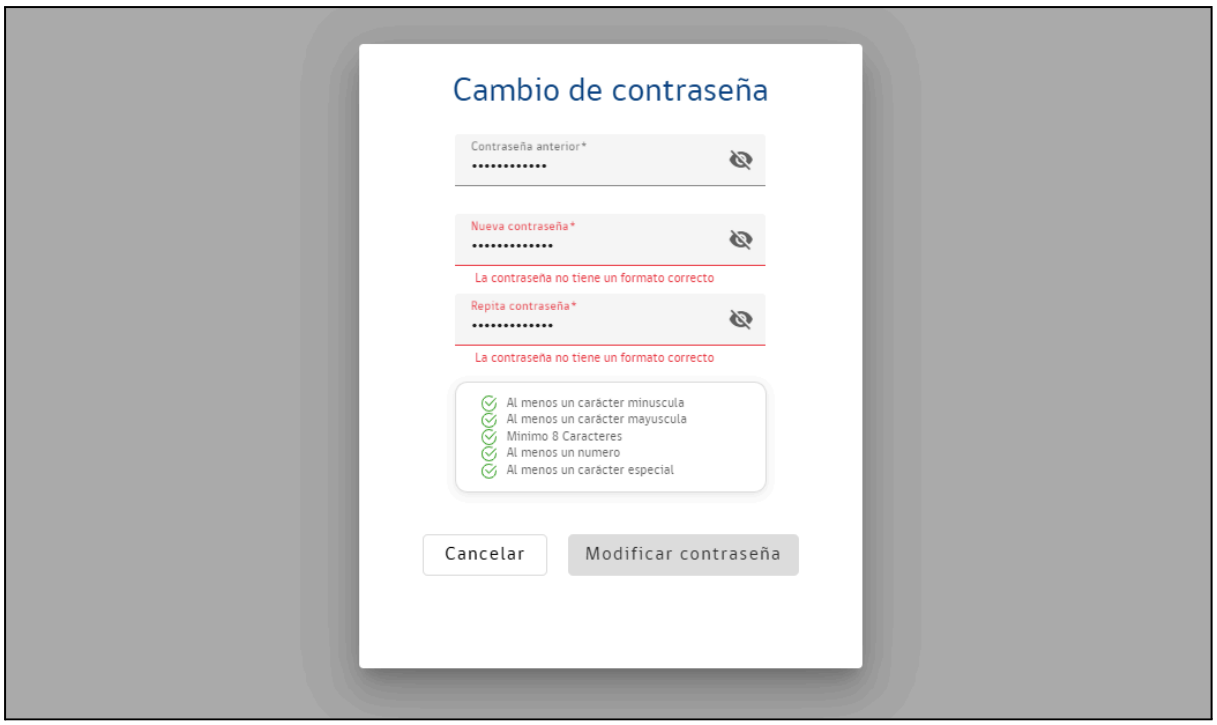
Para el caso del login, solo el campo “Código entidad” se valida desde la web. Los campos “Usuario” y “Contraseña” se validan una vez se haga clic en el botón “Ingresar”. Ejemplo validación campo “Código entidad”:

3.1.2. Cambio de clave Login

Desde la opción de login “Recuperar contraseña”, y luego de recibir el correo con los datos necesarios para el cambio de clave web, si el usuario utiliza un valor no soportado por las reglas se mostrará como mensaje “La contraseña no tiene un formato correcto”:

3.1.3. Cambio de clave Menú PIAS

Una vez logueado en la aplicación web, desde la opción en el menú lateral “Cambiar contraseña”. De la misma manera que el cambio de clave Login, alerta a los usuarios sobre caracteres no permitidos con el mensaje “La contraseña no tiene un formato correcto”:



3.1.4. Creación de usuario (solo campo username)
Desde Administración > Usuarios > Nuevo usuario, si se ingresa un valor no válido en el campo “Nombre de usuario”, el sistema mostrará el mensaje “Carácter no permitido”: