Es momento de trabajar con validadores asíncronos que la verdad es exactamente lo mismo solo que tiene

una pequeñito cambio les voy a pedir que creamos un nuevo espacio para trabajar con un nuevo campo comienze

todos lo del correo.

Voy a pegar abajo abajo del mismo correo por lo que hemos aqui usuario que esta va a ser un usuario

explains Wolder ingrese su usuario o nombre de usuario mejor nombre de usuario ok encontró el correo

aqui vamos a poner usuario en el correo no ha habido aquí voy a cambiar esto por usuario usuario no

valido lo mismo que teníamos vamos a quedarnos este Guebel y hay que crearse el campo en el que también

podría cambiar esto ese nombre de usuario ya fue tomado vamos a poner una validación así entonces el

usuario no lo voy a tomar.

Voy a tomarme este kieren y lo voy a establecer en mi formulario justo abajo del correo para mantener

la misma estructura y aquí va a ser usuario no válido.

Aquí va a ser usuario que obviamente este campo todavía no existe y lo vamos a definir.

Bajemos un poco más.

Le voy a colocar abajo del correo que este usuario ustedes pueden definir.

Por cierto esto no lo expliqué pero si nos necesitan por ejemplo una validación síncrona entonces simplemente

van a poner una coma.

Espacio y ahora definan.

El tercer argumento del arreglo va a ser el validador asíncrono y ese validador asíncrono se va a llamar

de disputo lo vamos a colocar siempre en nuestro servicio validadores punto existe.

Usuario Ok voy a ponerlo de esa forma.

Ok no necesito mandar ningún argumento porque recordemos que una variación de este tipo lo que hace

es mandar todo el control y yo necesito únicamente el valor de ese control y así puedo definir si eso

sólo existe o no ya lo vamos a ver.

Ok entonces obviamente existe usuario no existe.

Voy a grabar los cambios acá y voy a definirme el método acá entonces se va a llamar usuario es existe

usuario y le pusimos lo vamos a definir acá.

Si este usuario abrís arrolladas cómo la otra validación de hecho es exactamente la misma validación

que tenemos acá con la única diferencia que en lugar de regresar esto regresamos una promesa que resuelve

eso.

Eso sería todo.

Entonces me voy a poner aquí dos puntos y esto va a resolver o una promesa promesa que voy a poner y

prometo yo lo voy a quitar o vamos a poner una tubería o un observado Tapp asegúrense importarlo sólo

para el tipado.

Qué resuelve ese mismo objeto.

Obviamente que me marcó un error pues todavía no lo estoy retornando a menos de que yo hago un retorno

a cero llaves.

En este caso voy a ponerle existe Trump OK pero obviamente aquí no estamos regresando a la promesa todavía.

Pero esto sería lo que se espera lo que vamos a hacer para no poner Veny y para no tener eso tan raro

entonces voy a quedarme aquí arriba una pequeña interfaz solo para no poner mi interfaz Spazio se va

a llamar error validad valid.

Bueno para que ustedes quieran que no lo estoy exportando porque sólo quiero que funcione en este archivo

y podemos copiarnos aquí.

Este objeto sin las llaves y la voy a colocar aquí adentro.

Pocas palabras dice que esto puede regresar cualquier cantidad de llaves iban a ser julianos esas ya

es el valor de esos llaves va a ser Juliano.

Ahora puedo tomar este error Validad y cambiar este Yeny por el error validez.

También puedo cambiar esto todos esos jeto por error validez y lo mismo podríamos poner aquí abajo.

De hecho lo podríamos poner simplemente lospuntos error.

Ah bueno pero aquí es diferente porque no regresamos nada simplemente se lo establecemos al formulario

si aquí no va perfecto.

Obviamente este error no existe no está bien porque no estamos regresando a la promesa.

Qué es lo que vamos a hacer.

Primero voy a definirme mi promesa constante de promesa aunque aunque voy a ser retraigan porque no

necesito almacenar la variable puedo ser todo en una sola línea y esa línea.

Entonces New Promises parentesis punto y coma.

El programa recibe su rezó como.

Aunque el reject no lo vamos a utilizar pero es conveniente por si acaso se hace una petición HTTP y

quiere manejar el error.

Pero si algo saliera mal también sería el mismo rizó que vamos a tener que trabajar con él rizó vamos

a hacer algo asíncrona con un set time out TAP.

Vamos a hacer que se resuelva en tres segundos o tres segundos y medio para que nos dé chance de ver

y esto en teoría es algo que se ejecuta a destiempo.

Básicamente sería nuestra petición asíncrona que ustedes pueden hacer un lanzamiento una petición HTTP

a cualquier cosa y lo que voy a hacer acá es que voy a preguntar si el control que por cierto aquí recibimos

también el mismo control que tenemos aquí abajo eso es lo que recibimos en el existe usuario.

Puedo preguntar si el control apuntó Balio sería lo que la persona escribió es exactamente igual a lo

que digamos que veamos que escribió extraida.

Si el usuario escribió extraida voy a decir que es usuario ya existe por lo cual voy a marcar un error

y para marcar ese error va a ser el Whishaw paréntesis a los cero llaves y voy a marcar existe como

truco.

Si ese usuario no existe entonces qué es lo que debemos de regresar o saber si se acuerdan que regresamos

revisamos Null exactamente igual como lo teníamos acá.

Esto quiere decir que la delegación pasa si la validación resuelve un objeto de este tipo quiere decir

que está chocando esa validación.

Voy a grabar los cambios y recuerda que esta validación se hace en tres segundos.

Esto es básicamente todo loscambios.

Déjeme revisar que tengamos puesta esta validación.

Note que quiere poner esta validación asíncrona por eso estoy dejando mi espacio acá.

Aquí serían los validadores síncronos.

Pero bueno dejarlos grabar los cambios.

Vamos a revisar.

Quiero que prestemos atención prestamos atención acá a el estado del formulario.

Pongamos unas contraseñas sólo para enfocarnos en lo que quiero y en los valores de los valores del

paso 1 1 1 2 3 2 1 2 3.

Lo único que quiero que nos enfoquemos en esta parte del usuario y también en el pendían ok.

Por qué salió pendían en el momento en que yo recarguen porque se disparó la validación asíncrona aunque

no se debería de disparar porque todavía no he escrito nada pero en el momento en que yo empiezo a cambiar

sus valores del usuario por ejemplo 1 2 3 noté que hice pendían tres segundos y medio después para responder

válido porque este no es el usuario que estoy bloqueando.

Recuerde que la única el único usuario que digamos que tengo registrado sería Strait.

Si yo pongo extraida dice pendiente 3 segundos y medio después aquí me dice falso y aquí tenemos el

error de que ese nombre de usuario ya fue tomado de ahí.

Ustedes pueden poner una mano a determinar si la validación que está chocando es el que ya existe o

es el recuadro o lo que ustedes quieran pero de esta manera funciona la validación es asíncronas es

exactamente lo mismo con la única diferencia que resuelven o una promesa que resuelve mi objeto que

tengo acá o un observado que resuelve ese mismo objeto pero son exactamente iguales las dos validaciones

la asíncronas y las asíncronos y recuerde no se lo voy a olvidar que la validación es asíncronas.

El tercer argumento que se envía en la configuración de mi control y antes de terminar esta clase sólo

quiero arreglar el problema que tenemos que cuando se recarga el formulario está disparando mi petición

y yo realmente no quiero eso sólo quiero que se dispare la petición cuando el usuario escribe algo porque

caso contrario sería una validación síncrona y ahí podríamos atajar ese ese problema de que sea obligatorio.

Entonces lo que se me ocurre es que podemos ir a la parte de los validadores nuestros valedores servis

y aquí adentro de ese existe usuario antes de regresar este New programes puedo hacer una pregunta y

decir si el control punto valio sea si no existe ese valor entonces voy a hacer un retorno de un punto

Resolve.

Esto es como una promesa resuelta inmediatamente y voy a resolver Null.

Quiero decir que todo está bien.

Digamos que esta validación sólo se encarga de evaluar el valor del campo no de que esté lleno o no

aunque también lo podrían hacer aquí si quieren pero me gusta que la validación sólo haga lo que se

lee lo que se supone que debe de hacerse realidad si existe el usuario o no.

Ok voy a poner aquí en el caso de que no existe ningún valor no estamos acá.

Voy a recargar el navegador web que dice inmediatamente Truc porque el campo no es obligatorio pero

si empiezo a escribir por ejemplo estribe ese campo es bueno no es que esté bloqueado de manera Tru

escribe extraida.

Vas a ver que la validación entra y ahí está evaluando el nombre perfecto y ahí me dice que el usuario

ya existe extraido 2 por ejemplo ese sí debería de pasar tres segundos y medio después tenemos que ese

sí es válido.

Ok eso sería todo lo que quería cubrir por la parte de los validadores asíncronos no es diferente que

esos validadores como ustedes pueden observar.

Lo único que cambia es que resuelven una promesa o observable si los veo en la próxima clase.