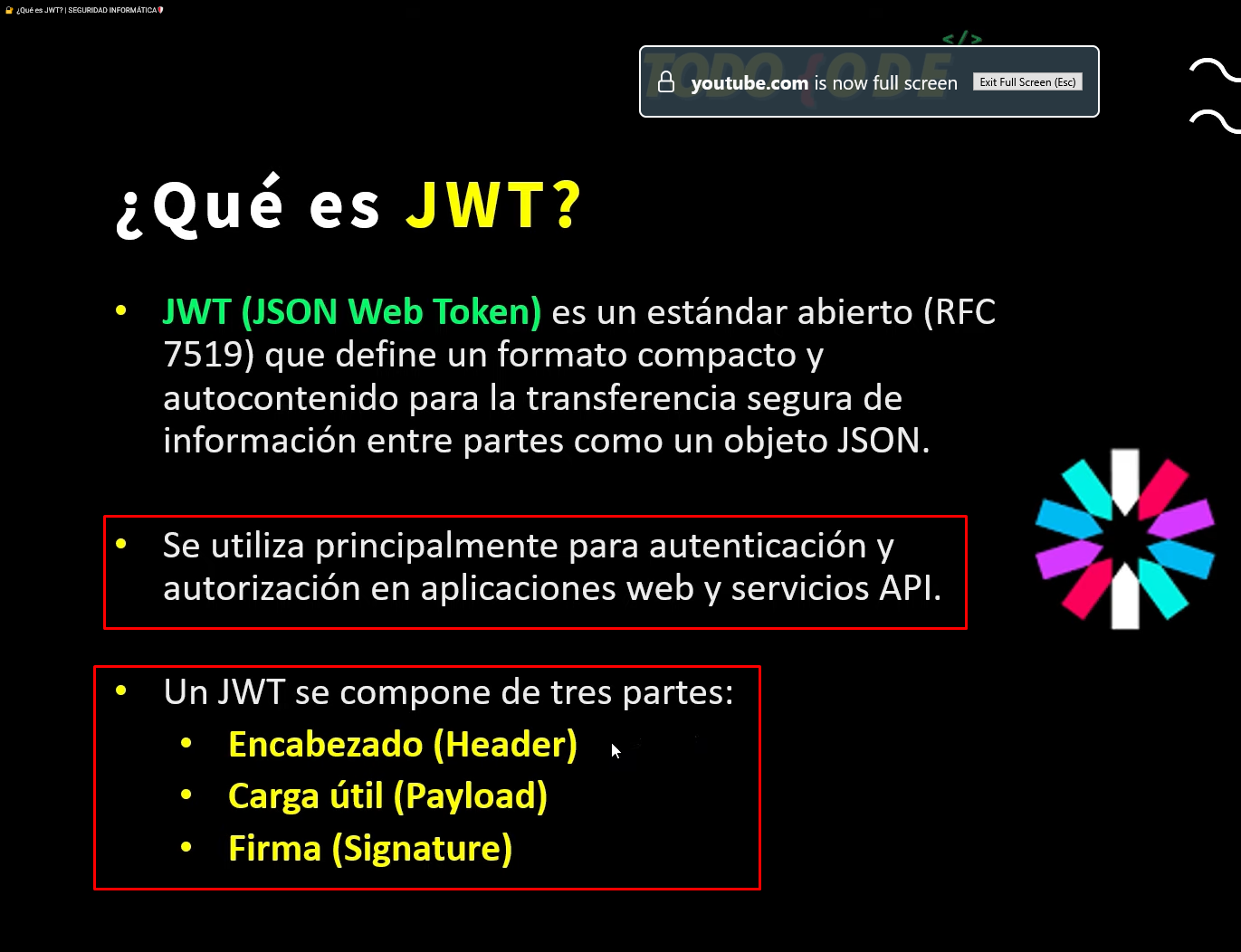
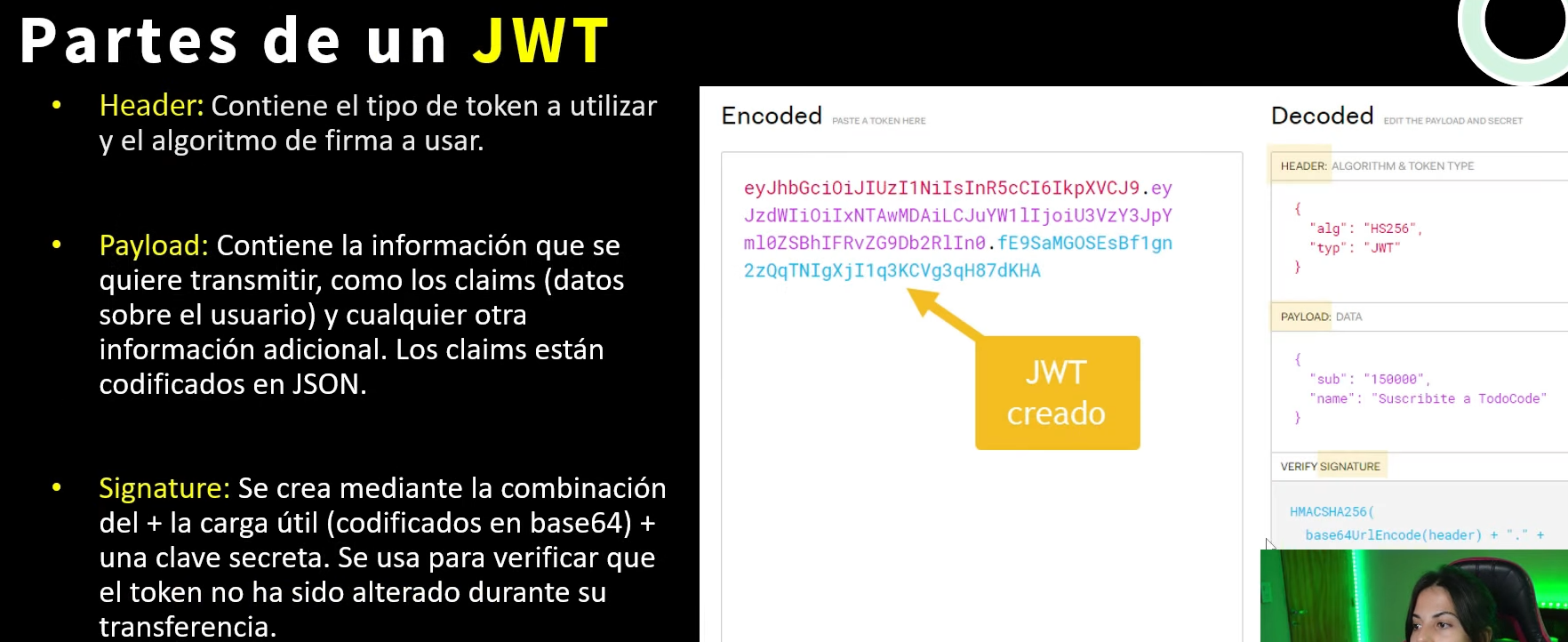
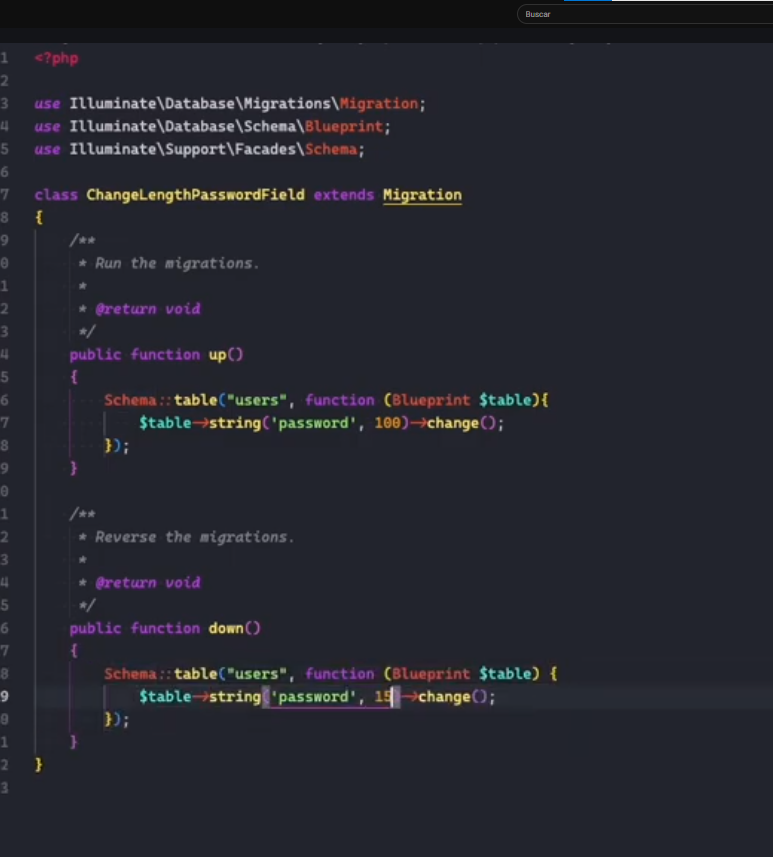
La **tokenización**, cuando se aplica a la seguridad de los datos, se refiere al proceso de sustitución de un elemento de datos sensible por un equivalente no sensible denominado token, que no tiene un significado o valor extrínseco o explotable. El token es una referencia (un identificador) que regresa a los datos sensibles a través de un sistema de tokenización.

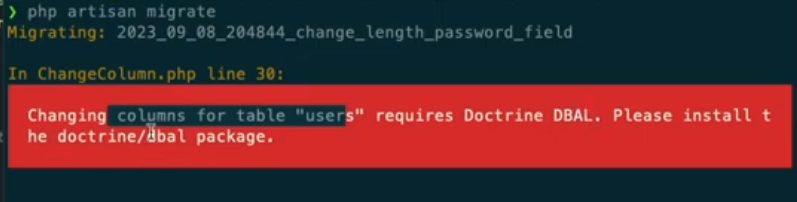




me está diciendo que se tuvo que truncar Y esto es aquí mismo me dice que es porque la Data es muy grande en la columna password Entonces qué tendría que hacer lo primero que tendría que hacer sería que necesitaría hacer una nueva migración en donde yo pueda expandir justamente el tamaño de el password en este caso por ejemplo ahorita lo tengo de tipo 15 luego voy a meter por ejemplo a tamaño 100 Entonces vamos a hacer algo vamos a crear aquí justamente dentro de nuestra terminal voy a parar el servidor un momento y le voy a decir

php artisan make:migration change\_length\_password\_field





CÓMO HACER peticiones POST con Laravel Lumen y validaciones en tu API. - YouTube

https://www.youtube.com/watch?v=C3NN2gUvu0I

Transcript:

(00:00) vamos a empezar por generar una nueva ruta ya en el vídeo pasado pudimos generar la ruta de login y pudimos empezar a generar la primera ruta que era la ruta get para todos los usuarios y le agregamos un middleware que en este caso se llama jwt Auto [Música] Bienvenido a mi canal Aquí encontrarás información que te ayudará a desarrollar tus habilidades dentro del desarrollo web Entonces lo siguiente que vamos a hacer es que vamos a agregar una ruta podemos duplicar la que teníamos y le vamos a agregar una ruta del tipo post

(00:32) en este caso el en Point como tal puede ser el mismo pero ahora vamos a hacer uso del user controller solamente que en el método Stone una vez que lo guardamos Entonces tenemos este user controller y dentro de user controller podemos empezar a trabajar en esta función o en este método que se denomina Store ahora lo primero que tenemos que saber es que tenemos que devolver algo por lo tanto vamos a retornar una respuesta en este caso va a ser un Jason y le vamos a agregar el request porque lo que quiero que observen es que este método rico es

(01:05) en realidad tiene un montón de información que normalmente traemos Entonces entonces si nosotros hacemos una petición justamente al enpom que acabamos de desarrollar aquí podemos observar que tenemos un montón de información que trae pero pues bueno lo que nos interesa es justamente los atributos que nosotros le estamos enviando para esta parte Entonces estos atributos se van a enviar mediante el método o mediante el objeto rico sin embargo adicional a eso nosotros tenemos justamente un modelo que ya está prediseñado sino recordamos pues tenemos

(01:41) un modelo users en donde tenemos todos los datos Así que podemos hacer uso de ese modelo user y le podemos poner que usert va a ser igual a New user que en este caso este user es el modelo que Nosotros acabamos de generar en los videos anteriores Y una vez que ya tenemos esto pues entonces le puedo decir que user en el campo name va a ser igual al request en el dato name podemos hacer esto para todos los demás por ejemplo para el email podemos hacer esto para el font podemos hacer esto para el password y podemos hacer esto para el

(02:18) estatus y una vez que tengamos esto dentro del modelo pues entonces le podemos decir al modelo que user se con esto lo que haríamos es que guardaríamos los datos Entonces observen lo siguiente Aquí vamos a empezar justamente a trabajar con algunas cositas importantes me voy a pasar de nueva cuenta al insomnia y dentro del insomnio lo que le voy a decir es que voy a enviar este set de datos recordemos que por defecto en nuestra base de datos nosotros teníamos que el campo usen es de tipo Unique el campo email Perdón es de tipo Unique por

(02:55) lo tanto aquí nos debería de empezar a generar nuestro primer error porque si yo le dejo el mismo campo pues me debería de dar ahí algún error y como pueden observar aquí me dice que tengo una violación de integridad y esto es justamente porque el email tiene que ser único Entonces qué podría hacer por el momento le voy a dar un uno simplemente y aquí en este caso observen que ya retorna justamente de nuevo el request Esto fue porque no lo cambiamos pero si nosotros observamos aquí pues ya tenemos justamente nuestro registro realizado

(03:30) ahora este registro realizado necesitamos agregarle algunas cosas importantes Por qué necesitamos agregarle unas cosas importantes Porque si observan por ejemplo el password está en texto plano y esto pues obviamente debería de ser un error si es que nosotros lo ponemos de este tipo y adicional a eso pues nosotros podríamos validar cada uno de estos datos ejemplo si yo por ejemplo le pongo vacío y le pongo dos observen que me lo guarda vacío cuando en este caso el nombre por ejemplo debería de ser requerido Entonces vamos a empezar por trabajar

(04:04) con eso lo primero que le vamos a hacer es que le voy a poner o le voy a generar un request Perdón un user para que nos mande el usuario que acabamos de trabajar y antes de poder hacer toda esta información para poder guardar registros vamos a empezar a validar justamente esto lo primero que tenemos que hacer es que vamos a poner un dis y este Dish le vamos a agregar el método validay este método validate necesita dos parámetros importantes el primer parámetro es el rico es como tal y el segundo parámetro es la validación que

(04:37) nosotros le vamos a hacer entonces por ejemplo para el campo name lo que necesito es que el campo name sea requerido y le vamos a poner este y le vamos a decir que sea de tipo String para el email le voy a decir que sea requerido que sea de tipo email pero que además sea único y un Nick y le tengo que poner dos puntos y le tengo que mencionar a partir de qué tabla quiero que trabaje con ese campo y Unique en este campo En este caso la tabla con la que es Unique es con users si nosotros nos vamos a la base de datos nos damos

(05:12) cuenta que la tabla se llama users por eso le pongo users acá y por último también le voy a decir por ejemplo el font que sea y adicional que tenga 10 dígitos entonces Dj 10 Si crees que mi contenido te aporta valor te pido por favor que lo compartas y me dejes un buen like ya que esto ayuda a su posicionamiento bien De dónde saco toda esta información de dónde encontré todos estos datos de recuadro String y todo lo demás si nosotros googleamos o buscamos laravel Lumen validation vamos a poder encontrar esta ruta o vamos a poder encontrar esta

(05:51) parte y aquí justamente nos dice que el la documentación full se encuentra en esta parte del árabe y en la parte del árabe aquí podríamos revisar todo lo que tenemos pero lo importante o lo que nosotros nos nos toca empezar a trabajar es la parte de cuáles son las validaciones que necesitamos Entonces entonces aquí justamente tenemos una sección que se llama a y Label validation rules en donde tenemos cada uno de las cada una las reglas que podemos ocupar aquí se encuentra el require aquí dicen por algún lado se

(06:24) debe de encontrar el String etcétera etcétera cada uno el digital también aquí está etcétera etcétera entonces pues de ahí sacamos todos esos datos del Unique también debe estar por acá justamente en algunas en algunos casos y aquí está justamente entonces pues bueno justo con esto lo que estoy haciendo es que estoy validando y ahora de nueva cuenta si yo empiezo por ejemplo con estos datos aquí si le doy Send observen que me dice que el nombre es requerido y adicional también me dice que el email ya se ha tomado adicional en este caso

(06:59) el teléfono es de 10 dígitos pero si le pongo algunos otros más entonces me añade que el teléfono debe ser de 10 dígitos y ya con esto hemos realizado la validación de nuestro de nuestro en Point de tipo post para la inserción de datos Ahora les decía que la parte del password pues es una parte en donde nosotros tenemos que agregar una validación extra Porque si recordamos para este caso en por ejemplo el password viene de tipo texto plano y no debemos de No deberíamos de guardar los datos de tipo texto plano para eso existe una una

(07:36) clase y dentro de la clase existe un método que se denomina Hash Entonces yo aquí le puedo poner Hash punto dos puntos make y le puedo decir que con esto haga el uso de Hash es importante que observen que aquí automáticamente ya lo tengo agregado pero viene de illuminate supports Entonces lo tenemos que agregar y una vez agregado ya podemos hacer uso del hashmake entonces con esto en teoría si yo pongo por ejemplo de nueva cuenta Mis datos correctos Aquí le voy a poner tres por ejemplo observen lo siguiente observen esto que me está pasando ya se

(08:17) está insertando se está haciendo la correctamente sin embargo me está diciendo que se tuvo que truncar Y esto es aquí mismo me dice que es porque la Data es muy grande en la columna password Entonces qué tendría que hacer lo primero que tendría que hacer sería que necesitaría hacer una nueva migración en donde yo pueda expandir justamente el tamaño de el password en este caso por ejemplo ahorita lo tengo de tipo 15 luego voy a meter por ejemplo a tamaño 100 Entonces vamos a hacer algo vamos a crear aquí justamente dentro de

(08:52) nuestra terminal voy a parar el servidor un momento y le voy a decir

php artisan make:migration change\_length\_password\_field

para que sea declarativo que estamos por cambiar el password Field y aquí automáticamente Nos está dando justamente una nueva migración en donde ya tenemos la y la parte del down ahora en esta parte le tenemos que decir que vamos a hacer uso de la clase esquema y le vamos a decir que la tabla que se llama usuarios users va a tener una función que trae el Blue print que

(09:37) en este caso le voy a decir Table y este blueprint de tipo Table le voy a decir que va a tener un tablet de tipo String que va a ser el password y aquí es donde le voy a decir 100 y le voy a dar un método que se llama change ahora si yo quisiera que en algún momento se hicieron rollback pues bueno simplemente le podría poner acá lo mismo y mandarlo a 15 o en este caso simplemente lo podría bajar dejar vacío cualquiera de las dos es correcto Entonces observen lo siguiente Parece que ya todo está bien Sin embargo si yo le doy php artisan

(10:15) migrate para poder ejecutar la migración se darán cuenta que justamente me dice que para poder cambiar las columnas necesito de doctrin dival Entonces doctrin dival es justamente una dependencia que yo tengo que instalar con composer para poder empezar a trabajar con esto así que le voy a decir con Bowser require doctrina doctrin dival Entonces ya le doy instalar espero a que termine la instalación y una vez que ya está hecha Pues ahora puedo volver a ejecutar mi migración y ahí ya me permite migrarlo bien Si nosotros nos

(10:52) regresamos y actualizamos nos podemos dar cuenta que la longitud cambió de 15 a 100 por lo tanto si yo le doy aquí Send en teoría a Déjenme volver a agregar mi servidor aquí ya me debe de dejar justamente bien y si yo reviso pues podemos observar que ahora mi password ya está encriptado esto es todo por el momento nos vemos hasta la próxima

# Construye una API CRUD con GET, DELETE y PUT con Laravel Lumen



Construye una API CRUD con GET, DELETE y PUT con Laravel Lumen - YouTube

https://www.youtube.com/watch?v=NtIbD9GSZB8

Transcript:

(00:00) ahora podemos empezar a trabajar con los demás métodos para esto para poder agregar los demás métodos pues vamos a empezar justamente a crear cada una de las rutas que normalmente vamos a ocupar a la hora de estar trabajando a la hora de estar trabajando con nuestro nuestra Epa [Música] Bienvenido a mi canal Aquí encontrarás información que te ayudará a desarrollar tus habilidades dentro del desarrollo web entonces bueno dentro del web punto php que se encuentra en la carpeta users en la carpeta rouse Perdón routes web

(00:37) punto php Aquí vamos a ir agregando algunas rutas más qué necesitan qué otras rutas necesitamos Pues necesitamos una ruta get que se llame users y que le podamos pasar un ID bien ahora cómo obtengo este ID no se preocupen acuérdense que tenemos un método rico tenemos una clase rico es Entonces si nosotros nos vamos al show observen que aquí tenemos ID justamente y este ID es el que se proporciona justo en esta parte entonces Aquí le voy a decir que este va a ser el método show entonces aquí ya tenemos nuestra primera ruta

(01:11) ahorita vamos a ver cómo trabajamos con esto el siguiente que le podemos dar por ejemplo es uno igual que se llame pero que ahora sea de tipo delete y le vamos a decir destroy de dónde saqué este del mismo controlador si nos vamos hasta abajo aquí tenemos destroy qué otra ruta tengo que generar pues tengo que generar una de tipo put que en este caso se llame update Entonces ya tenemos para la ruta para todos los datos la ruta para un único dato la ruta para eliminar datos y la ruta para editar datos entonces pues vamos a empezar a trabajar

(01:47) con cada uno de ellos para el show para el show vamos a hacer lo siguiente de nueva cuenta vamos a poder tener por ejemplo un user y le vamos a decir que user viene del modelo user y le voy a dar dos puntos web es un método en donde nosotros podemos Buscar condicionadamente dentro del modelo users Entonces le vamos a decir que Where ID que en este caso es el nombre de la columna que queremos Buscar Recuerden que este ID pues justamente es este nombre de columna y le vamos a decir que sea igual a al aire y una vez

(02:22) que ya lo tenemos Podemos agregar el método games para poderlo obtener Entonces ya tenemos esta parte ahora qué debemos de hacer pues una vez lo que debemos de hacer ya lo único es que le damos retum y le damos avispoints que en este caso va a responder que el usuario user y con esto en teoría debería estar trabajando si nosotros nos vamos a insomnia y aquí ya tengo el get user observen lo siguiente Voy a agregar un ID válido Como por ejemplo el uno si le doy aquí le doy sets Ah bueno me dice que mi toque en expegado vamos a agregar

(02:58) un nuevo token Aquí está entonces si le doy Send aquí ya me trae el dato pero ahora qué pasa Qué pasa si por ejemplo le doy el número 2 que no lo tengo vamos a ver eso Pues aquí me lo manda vacío Sale pero si le pongo otro adicional por ejemplo el 20 Pues aquí me lo trae vacío qué podría hacer Pues yo podría hacer una condición por ejemplo aquí le puedo decir count the user es menor que uno entonces le puedo decir que me retorne que me retorne el response de tipo Jason y que traiga un dato que diga error y en este

(03:40) caso user not y listo Entonces ya con esto por ejemplo si no lo tengo Pues acá me dice error user not eso ya tenemos para poder empezar a trabajar con cada uno de nuestros datos de manera individual si crees que mi contenido te aporta valor te pido por favor que lo compartas y me dejes un buen like ya que esto ayuda a su posicionamiento ahora si lo que yo quisiera hacer es actualizar los datos pues actualizar es muy muy similar a lo que tenemos dentro del Store Así que incluso me voy a copiar toda esta información dentro del Store

(04:18) que tengo aquí y lo voy a publicar dentro de update y simplemente le hacer algunos cambios hitos en principio le voy a decir validate request y que siga validando pero ahora lo único que le voy a decir es que el email no va a ser de tipo Unique Por qué Porque lo que quiero es que puede ser que hayan actualizado pero puede ser que no hayan actualizado justamente Entonces si no han actualizado los datos o si no han actualizado el email pues no me no me no me sirve que me aparezca La Leyenda por eso lo voy a eliminar y aquí lo único

(04:53) que tengo que hacer es en lugar de New y user porque no creo que no quiero que sea un nuevo objeto quiero que venga de user y Exactamente lo mismo que hicimos en la parte de arriba le vamos a decir Where ID sea igual al ID y que me traiga el primero First y listo con esto lo que estamos haciendo es que vamos a obtener Cuál es el valor o cuál es el registro que le estamos trayendo por ID entonces por eso es importante que en la ruta en esta ruta hemos dejado este ID entonces aquí vamos a traer el ID y una vez que ya traigamos

(05:28) el ID pues empezamos a con todos los datos que tenemos por acá y al final simplemente lo salvamos porque lo salvamos Pues porque Recuerden que en realidad estamos editando un registro Sale entonces a diferencia de en el estor en donde creamos un nuevo registro en este caso lo estamos salvando porque ya actualizamos el registro que tenemos para acá y simplemente pues le damos guardar Así que yo puedo hacer por ejemplo acá le puedo decir update user le puedo decir uno y que me empiece a actualizar el uno el token expirado

(06:03) Déjenme actualizarlo y aquí como podemos ver se va actualizando de hecho por ejemplo si yo le pongo a todo un uno para poder verificar por ejemplo Aquí le voy a poner un 11 aquí nos da algún error vamos a ver qué es lo que pasa y nos está diciendo que el uno se está violando pero observen que se está violando justamente porque ya tenemos un uno entonces aquí le podemos dejar por ejemplo el cero y esto ya se está actualizando si le damos aquí actualizar pues ya no lo trae y si lo eliminamos todo como estaba antes pues nos sigue

(06:37) funcionando y nos sigue actualizando correctamente ahora ya tenemos un date ya tenemos show ya tenemos Store nos hace falta simplemente empezar a eliminar el registro y para poderlo eliminar la verdad es que está bastante sencillo ya sabemos Cómo consultar el usuario Entonces le voy a decir que nos traiga la consulta del usuario y una vez que nos traiga la consulta al usuario simplemente le digo usert delete y con esto le puedo dar que me responda en formato Jason y por ejemplo que me diga mensajes o Data y aquí podemos user with

(07:17) ID y le podemos concatenar el ID del lettery aquí Me faltó todo entre corchetes porque acuérdense que es un arreglo Perfecto entonces con esto ya tenemos lo mismo observen que si le ponemos vamos a eliminar por ejemplo el 5 si le ponemos vamos a eliminar el 4 Entonces le ponemos cuatro y Déjenme pasarme la ruta delete y vamos a poner el 4 entonces le damos acá me dice que el token ya expirado lo copio otra vez lo pego en mis encabezados Send y aquí me dice que el user con el del con el 4 ya ha sido eliminado no pero qué pasa si

(07:56) lo vuelvo a poner acá pues observen que me aparece un internal error Por qué Porque la función de lidiana la encuentra ya no encuentra la del nulo Entonces qué tengo que hacer pues simplemente puedo hacer una validación antes del delete y le puedo decir que If Y en este caso user pero la vamos a hacer negativa o negada Entonces le vamos a decir si no se encuentra el usuario entonces podemos retornar un response en formato Jason que diga error y aquí user not Funk y ya con esto lo que estamos haciendo es que en dado caso

(08:32) pues ya nos manda el 200 pero en dado caso de que no exista y si sí existe pues simplemente los lo elimina y aquí podemos actualizar y podemos observar que nuestros registros ya están cargados esto es todo por el momento nos vemos hasta la próxima

# Crea una API REST de Categorías con Laravel Lumen: GET y DELETE

Crea una API REST de Categorías con Laravel Lumen: GET y DELETE - YouTube

https://www.youtube.com/watch?v=s0aFhg1lGVA

Transcript:

(00:00) Ok entonces Vamos a continuar de nueva cuenta con nuestro curso para esto vamos a crear otro controlador y el controlador va a ser el controlador de categorías.

entonces pues bueno vamos a empezar justamente por trabajar nuestras categorías Así que [Música] Bienvenido a mi canal Aquí encontrarás información que te ayudará a desarrollar tus habilidades dentro del desarrollo web Le voy a dar el comando php artisant y le voy a decir make controller make model y le voy a dar algunos atributos para poder trabajar en principio le voy

(00:35) a decir menos C para que sea la Api Además de eso también le voy a decir r para que sea el recurso Es decir para que automáticamente me mande todos los métodos que trae justamente el el controlador y le voy a decir m para que me cree también la migración entonces con esta con este flag o con este con estos Flags lo creando es el modelo que en este caso nuestro modelo se va a llamar tegory y le estamos diciendo que categoría va a tener un controlador va a tener los recursos del controlador es decir los métodos ya creados y va a

(01:15) tener la migración Entonces vamos a darle enter y vamos a ver qué sucede Okay Al parecer nos creó entonces simplemente le voy a generar el controlador y el recurso A lo mejor la migración ya me lo está tomando Entonces vamos a ver por acá Así que ya tenemos un modelo que se llame categoría y pues bueno por acá todavía no nos crea los recursos Así que vamos a empezar a generarlo Entonces php artisand lo quería hacer todo en un comando pero pues si no nos deja le damos mail controller y le decimos en este caso que

(01:46) nos genere el controlador que se va a llamar categórico controller y menos r para el recurso y ahí ya nos da justamente nuestro controlador y ahora solamente para terminar le damos que php y le damos make migrations y le vamos a poner a esta migration create categories entonces aquí nos está dando un error en las en el en la migración vamos a ver qué es lo que sucede y ahí yo estaba dando un errorcito pero ya debe de funcionar vamos a ver ya ella nos genera bien la migración Ok Perfecto entonces ahora pues lo que tenemos que hacer es

(02:25) que tenemos que definir Cuáles son los campos que vamos a usar Recuerden que por defecto ya nos da el ID ya nos da el Time stamps y pues de aquí incluso también podríamos agarrar algunas cosas no como por ejemplo el name este lo podemos poner acá podemos crear un booleano en este caso se va a llamar type y le ponemos uno podemos generar un String para un color y le vamos a dar seis y también podemos generar un String para un icon y le vamos a dar 255 vamos a ver si con eso funciona ya aquí podemos poner justamente lo mismo y

(03:02) esquema y listo Entonces vamos a ejecutarla php artisan migrate y listo ahí se ha generado la migración vamos a revisar en la base de datos a ver cómo está y aquí ya podemos observar que ya tenemos justamente esto primero No ya tenemos justamente el la tabla de categorías entonces pues bueno vamos a empezar justamente a trabajar con las categorías eso nos vamos a ayudar del controlador que ya habíamos hecho del controlador de users Así que si ustedes recuerdan aquí tenemos la función Index que la función Index es la primera que

(03:38) nos devuelve y esta función Index acuérdense que todo esto ya se construyó con base en el flac r que le habíamos puesto y listo vamos a empezar a trabajar entonces acá le voy a copiar todo esto y lo único que voy a cambiar es user si ya no se va a llamar user sino que ahora se va a llamar categoris y ya no es user sino que ahora sería ca entonces aquí ya tenemos el controllers categoría controller Y este categórico controller tiene que venir de los modelos Así que podemos copiar el modelo también y lo podemos pegar aquí en

(04:12) categoría Perfecto entonces ya con esto ya creamos nuestro controlador ahora qué es lo que tenemos que hacer Pues bueno lo que tenemos que hacer es generar la ruta para poder exponer este controlador así que puedo generar justamente aquí mismo aunque podría ser que genere generar otro grupo pero si quiero aquí mismo por el momento puedo generar un diagonal categoris y decirle que haga uso del categórico controller y en su campo Index Entonces vamos a ver qué pasa con esto vamos a ver si funciona si crees que esta información te aporta

(04:49) valor te pido por favor que lo compartas y que nos dejes un buen like y un comentario ya que esto ayuda a su posicionamiento Aquí está mi insomnia este lo puedo duplicar y le puedo poner categories y aquí simplemente sería categorías vamos a ver me dice que no se puede conectar el servidor obviamente esto es porque no lo he levantado php artisant set y me dice que el token expirado Perfecto entonces vamos a loguearnos aquí me voy a loguear el correo o el password están incorrectos vamos a revisar okay Aquí voy a hacer

(05:24) una prueba porque justamente también puede ser algo que tengamos que cambiar le voy a poner contacto uno Aquí ya está el token Y eso es algo que tenemos que cambiar porque si recuerdan en realidad nuestro password no cambio Pero como ahora está encriptado hay que irlo mejorando con base en lo que tengamos Entonces eso lo pasemos más adelante por el momento nos vamos a ir a get old categories y aquí le damos el nuevo token Send y ahora tenemos categorías vacío porque justamente categorice en este momento pues no tiene absolutamente

(05:53) nada va Vamos a agregar un primer registro solamente para tener aquí y le vamos a poner por ejemplo que este registro se va a llamar comidas el color que por el momento le voy a dejar es 000 que sería como un que sería el negro en hexadecimal aquí en el icon le voy a poner comidas punto potapg y el tipo va a ser uno que va a representar los gastos entonces aquí en el created le vamos a poner Now y aquí también Now por el momento Entonces ya aquí ya lo tenemos Y si actualizamos Pues aquí ya tenemos Nuestra primer categoría

(06:29) Perfecto entonces aquí ya empezamos justamente a revisar que esto esté funcionando ya ya tenemos lo primero así que pues lo siguiente sería pasarnos también hacia nuestro router y pues bueno Esto lo voy a pasar para acá y de hecho voy a copiarme este show y lo mismo le voy a poner categorías y aquí Que me lleve al categol así que pues vamos a generarlo vamos nos vamos al user controller para que lo tengamos de referencia y aquí si recordamos tenemos el show aquí simplemente es hacer una consulta al modelo en este caso es el modelo user

(07:08) pero nosotros vamos a usar el modelo category con base en el aire lo obtenemos Y si existe pues entonces lo retornamos Así que tal cual lo podemos copiar y lo podemos pegar justamente en nuestra función show y simplemente cambiamos los datos no es user sino que es categoría y acordémonos que aquí en realidad es categoría aquí en el lado caso de que no esté pues le ponemos categorite y listo con esto en teoría ya deberíamos de tener De hecho aquí podemos poner get all One get One user Entonces lo duplicamos get One categoría

(07:44) y listo también le tenemos que cambiar el token Recuerden que el token Lo acabamos de cambiar así que get One category y vamos a hacer la prueba algo ha ocurrido al de codear El token vamos a revisarlo get One a Y esto es porque aquí categoris este toque Ahí está y ahí está perfecto y si le damos dos categoría not Found Perfecto entonces ya ahí ya tenemos nuestro error ya con esto ya empezamos a hacer el get y ya tanto de todos como el individual Perfecto entonces ahora que nos hace falta Pues bueno nos hace falta por ejemplo la

(08:23) función destroy la función distroll Recuerden que también es sirve para poder eliminarlo Entonces esto es muy sencillo aquí lo podemos empezar a generar lo copiamos tal cual de nuestro user controller que fue el primero que generamos y nos vamos a destroy acuérdense el modelo no es user es category y user aquí todo cambiamos todo por categoría con esto obtenemos el la categoría si no existe la categoría metemos un error que le diga que no hemos encontrado la categoría y en dado caso de que sí eliminamos la categoría y

(08:57) respondemos que la categoría ha sido el ID tal ha sido eliminada correctamente Entonces vamos a duplicar un registro por ejemplo este vamos a duplicar la fila para que sea el dos y yo aquí por ejemplo pues me puedo duplicar este mismo get y aquí delete y simplemente le cambiamos a delete y listo aquí podemos eliminar la categoría número 2 le damos ent y veamos Qué sucede nos dice que la que la ruta no está definida acuérdense que eso es importante porque pues lo tenemos que definir en esta parte de acá Así que la definimos le decimos

(09:35) categoris y acuérdense que es categoría y listo con esto en teoría deberíamos de tener definido y aquí ya nos dice que el 2 ha sido eliminado correctamente si lo volvemos a dar le decimos que la categoría no no existe Y esto es porque ya la eliminó esto es todo por el momento nos vemos hasta la próxima