-route: “/signin” method: ”post”

request:

body: {

email: string,

password: string

}

response:

{message: 'Ha ocurrido un error al encriptar tu contraseña. Intentalo de nuevo'}

or

{message: 'Te has registrado con exito! Ahora puedes iniciar sesion para empezar a usar nuestro servicio.'}

or

{message: 'Ha ocurrido un error. Intentelo de nuevo!'}

-route: “/login” method: “post”

request:

body: {

email: string,

password: string

}

response:

{message: 'El email o contraseña son incorrectos'}

or

{message: 'Ok'}

or

{message: 'Fallo al iniciar sesion'}

-route: “/forgotten-password” method: “put”

request:

body: {

email: string

}

response:

{message: 'Ha ocurrido un error. Intentalo de nuevo'}

or

{message: 'Correo enviado'}

or

{message: 'El correo no existe'}

or

{message: 'Error'}

-route: “/restore-password” method: “put”

request:

body: {

email: string,

password: string

}

response:

{message: 'Tu contraseña ha sido cambiada!'}

or

{message: 'El correo electronico no existe'}

or

{message: 'Ha ocurrido un error. Intentelo de nuevo!'}

or

{message: 'Error'}

\*Para las siguientes rutas el usuario debe estar autenticado, y usare fileForm si el tipo de dato es un archivo de formulario

-route: “/upload” method: “post”

request:

{file: fileForm}

response:

{message: 'OK'}

or

{message: 'error'}

\*fileKey es la clave del archivo en s3, sin el id del usuario (por ejemplo, si la clave en s3 es “4/nombre-archivo” entonces fileKey debe ser “nombre-archivo”

-route: “/download” method: “get”

request:

query: {

fileKey: string

}

\*oldKey en el mismo formato que menciono en el anterior comentario

-route: “/rename:file” method: “put”

request: {

oldKey: string,

newKey: string

}

response:

{message: 'OK'}

or

{message: 'error'}

\*api en la siguiente ruta seria una url, como “<https://i.imgur.com/F2iRSbO.jpeg>” o “<https://upload.wikimedia.org/wikipedia/commons/4/4a/100x100_black_and_white_pixels.png>”

A su vez, result.errors y result.response corresponden a la api de unsplash (en este caso el paquete de npm unsplash-js <https://www.npmjs.com/package/unsplash-js> )

result.errors seria:

“we return a result.errors object containing an array of strings (each one representing one error) and result.source describing the origin of the error (e.g. api, decoding). Typically, you will only have on item in this array.”

y result.response:

“we return a result.response object containing the data.

* + If the request is for a page from a feed, then result.response.results will contain the JSON received from API, and result.response.total will contain the [X-total header value](https://unsplash.com/documentation#per-page-and-total) indicating the total number of items in the feed (not just the page you asked for).
  + If the request is something other than a feed, then result.response will contain the JSON received from API

“

-route: “/upload:api” method: “post”

request:

body: {

api: string

}

response:

{message: 'OK'}

or

{message: 'Error'}

-route: “/image:search” method: “get”

request:

query: {

word: string

}

response:

{message: 'Error', data: result.errors}

or

{message: 'OK', data: result.response}