03/09/2018

Nodejs REST API

Metadata

La metadata le brinda la información necesaria a la API para realizar la validación individual y cruzada de los parámetros recibidos desde el cliente.

La API efectúa la validación individual de los parámetros respecto a tipo de dato, longitud y cantidad de decimales, y si el parámetro es requerido o no.

Efectúa también la consistencia de "uniqness" de las columnas, validación de existencia de valores de "foreign key" en insert y update, y chequeo de "primary key" como "foreign key" de otras tablas en delete contemplando, si así se indica, el "cascade delete".

Se denomina recurso a la unidad de afectación de la API. Normalmente un recurso tiene una relación biunívoca con una tabla de la base de datos. La excepción es el caso de una vista (view).

La metadata se compone de un grupo de archivos JSON (dentro de la carpeta "metadata" a partir del "root" de la aplicación). Existirá un archivo JSON por cada recurso. El servidor lee la metadata cada vez que se instancia. Al servidor se le debe indicar explícitamente que recursos debe leer (que archivos de metadata).

Estructura de un archivo de metadata:

{

resource: <string> nombre del recurso,

table: <string> nombre de la tabla en la base de datos,

verbs: <char array> [G,P,U,D] lista de los código de verbos permitidos sobre el recuso,

columns: <object array> array de objetos columna:

[

{

name: <string> nombre de la columna en la tabla,

rol: <string> [P, F, D, V] código de rol de la columna,

cascade: <string> [Y,N] si el rol es “primary key” controla la acción a realizar cuando se elimina un registro : si es “N” dará un error si tiene registros de otras tablas que lo referencian, si es “Y” eliminará, en la mediad de lo posible los registros de las tablas que lo referencian,

type: <string> [S,I,N,T,D,M,F] código de tipo de dato,

length: <integer> longitud de la columna aplicable a tipo de dato string como longitud máxima y para columnas decimales el largo máximo total del número incluyendo punto decimal y decimales (para el resto de los tipos de dato debe ser “null”),

decimals: <integer> cantidad máxima de decimales para el tipo de datos decimal (para otro tipo de datos debe ser “null”),

required: <string> [Y,N] si la columna es mandatoria (“not null”),

unique: <string> [Y,N] si el valor debe ser único en la tabla,

table: <string> para el caso de rol “foreign key” (caso contrario “null”) indica el nombre de la tabla cuya “primary key” es “foreign key”

}

]

}

Tablas de códigos:

Verbos:

Código	Valor
P	POST
G	GET
U	PUT

Roles:

P	Primary Key
F	ForeignKey
D	Dato
V	Version control (tipo integer)

Tipos de Dato:

S	String
I	Integer
N	Decimal
F	Float (o Double)
T	Datetime
D	Date
M	Time
B	Boolean

Al instanciarse, el servidor carga la metadata y valida la existencia de los recursos necesarios para efectuar el control de integridad referencial.