**TPE parte 3**

**Participantes:**

Ferrante Tomas Abel:

tomax19cr7@gmail.com

Romero Fabián Ignacio:

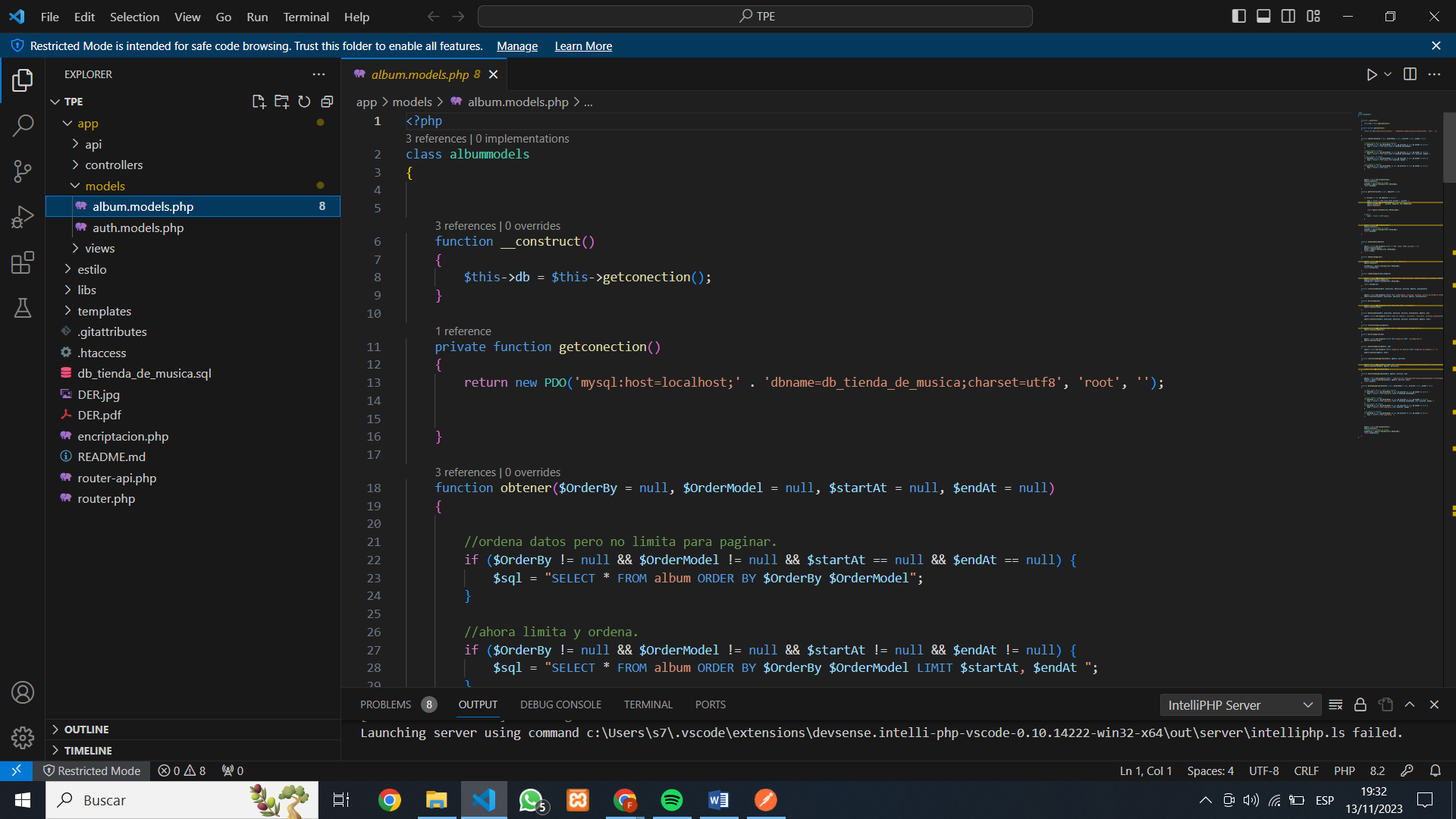
fabian.romero.ignacio@gmail.com

Documentación del api:

**Conexión con la base de datos:**

En la ruta models/album.models.php, ir al método getconection();

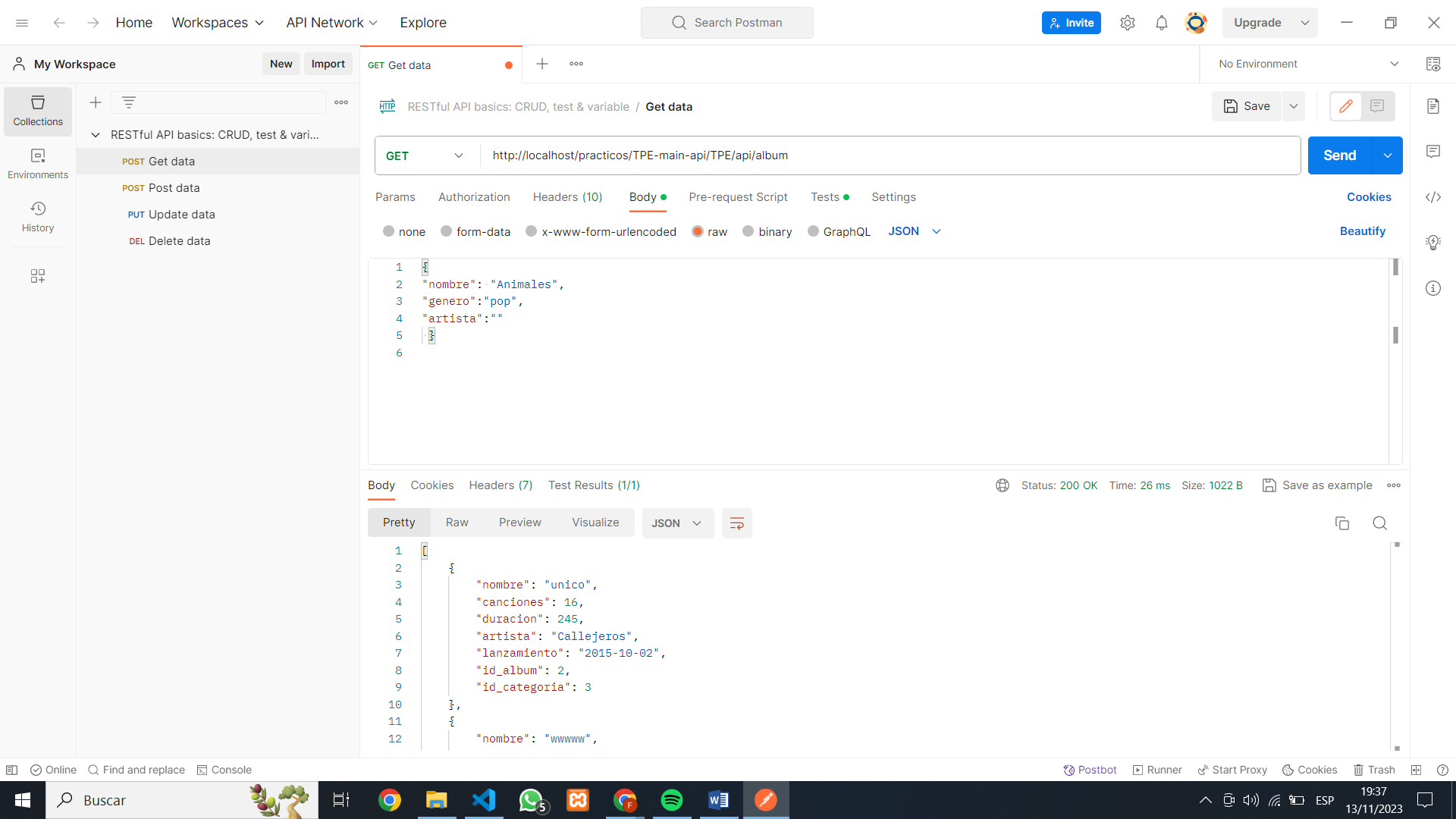
Aquí se puede configurar el nombre de la base de datos, el usuario y la contraseña. Root sería el usuario y en ‘ ‘ iría la contraseña.



**Solicitudes tipo get:**

Aclaración las solicitudes de tipo get están diseñados para el único uso de las tablas “album” y “sugerencia”, si usted tiene un fallo y no salta el error, verifique el nombre de las tablas.

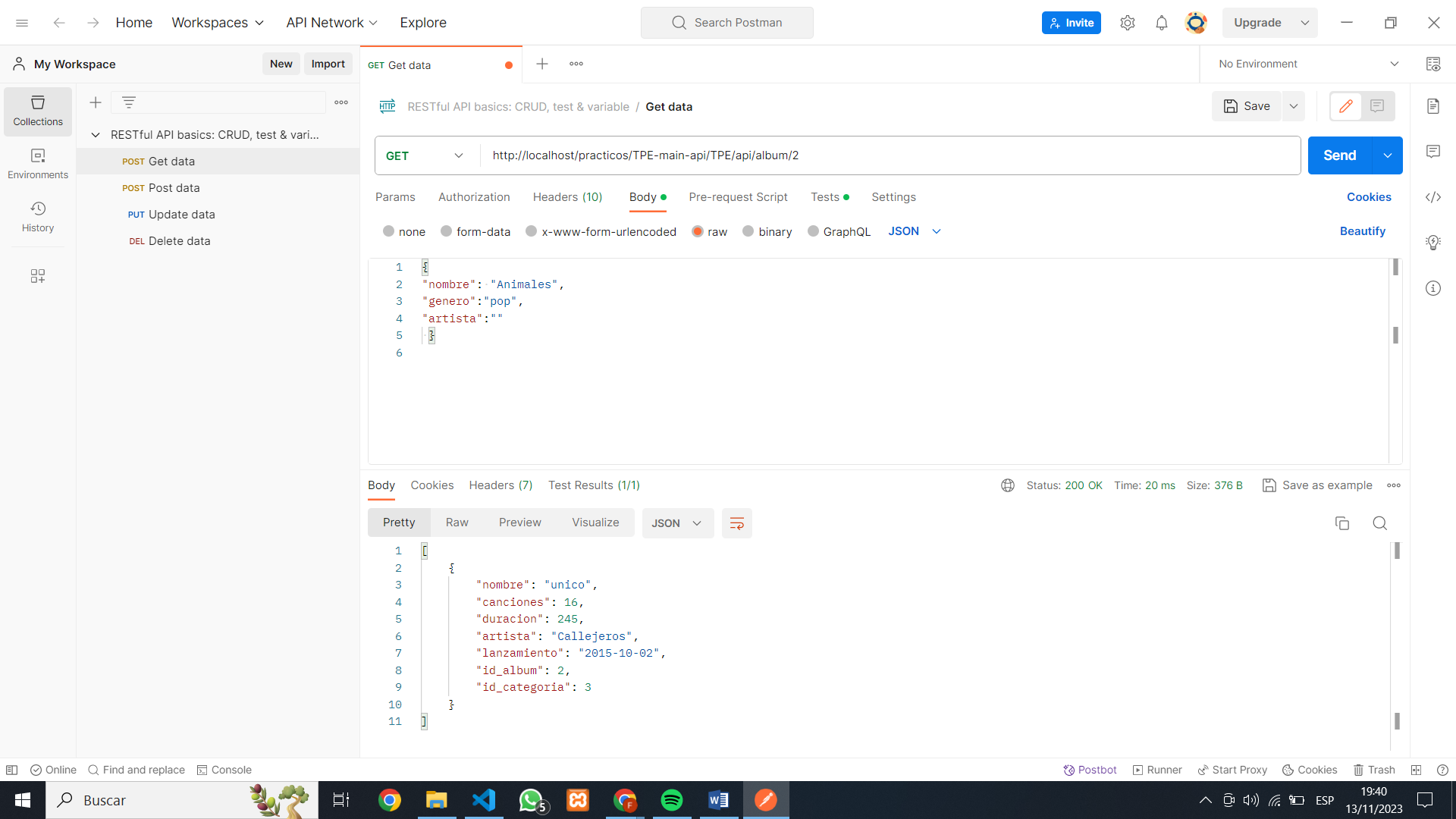
**Obtener todos los datos de una tabla**, para realizar esta acción simplemente hay que colocar el nombre de la tabla después del endpoint, en este caso álbum.



En caso de error de url devolverá un error 404.

**Obtener los datos de un álbum especifico,** para obtener un álbum en específico colocamos su id después de la indicación de que álbum queremos obtener ejemplo; El siguiente caso obtiene el álbum id 2.

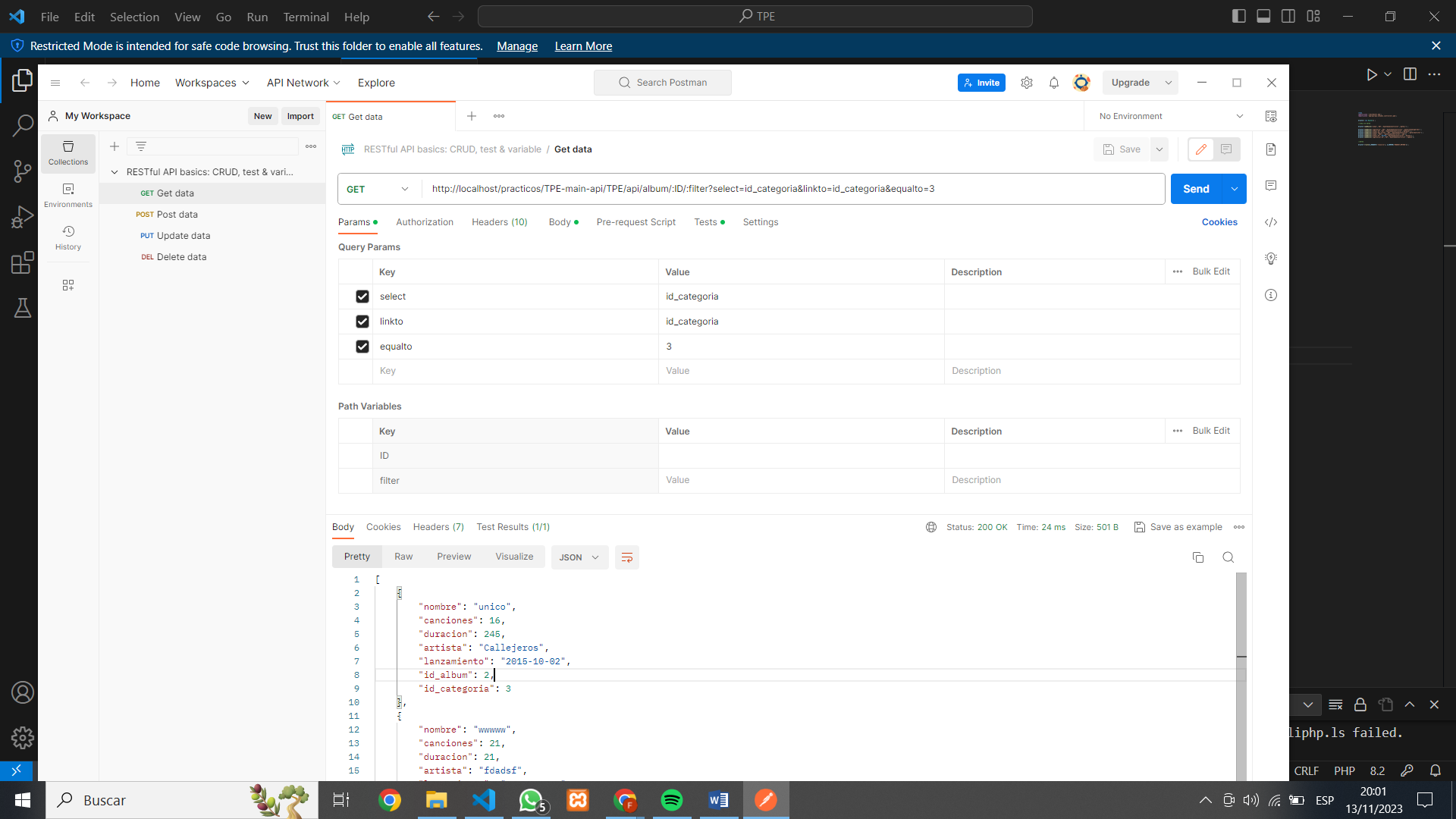
Solamente permitimos obtener datos específicos de los álbumes, es decir, **no será posible** buscar un **álbum sugerido** por el usuario atreves de su id.



Si el id que ingresaron no coinciden con ningún álbum se dará el error 404.

**Filtrado de sección,** agregamos los parámetros “linkto” y “equalto”, en el primero vamos a colocar la columna de la tabla, en el segundo parámetro colocamos el valor exacto de lo que queremos filtrar.

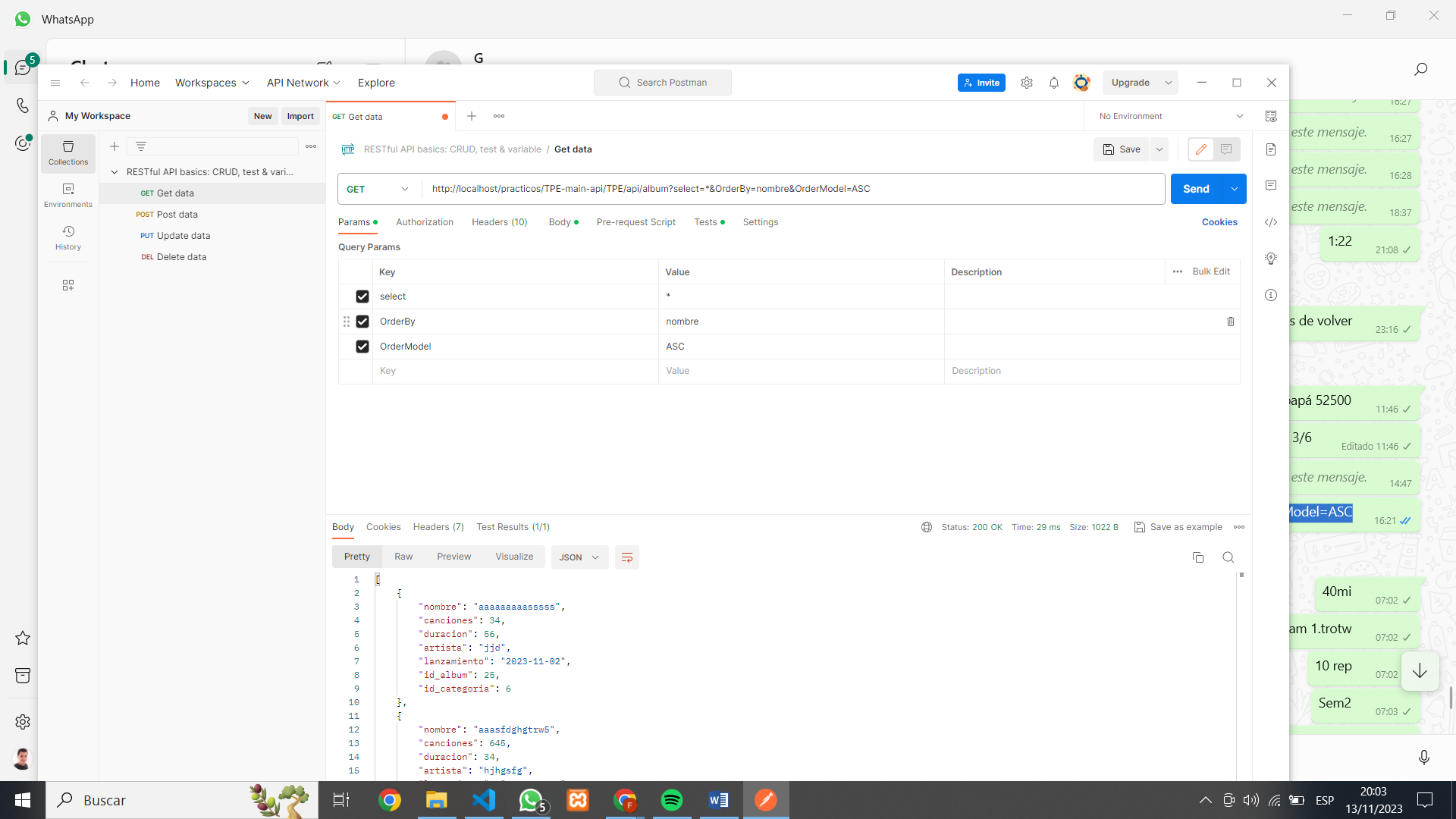
En este ejemplo filtre todos los álbumes con la el id\_categoria 3, en select



Si ingresan mal el linkto o equlto se dará el error 404.

**Ordenar datos,** agregando los parámetros OrderBy y OrderModel, en OrderBy indicamos la columna a ordenar y en OrderModel establecemos el tipo de orden con las palabras reservadas ASC(Ascendente) DESC(Descendente).

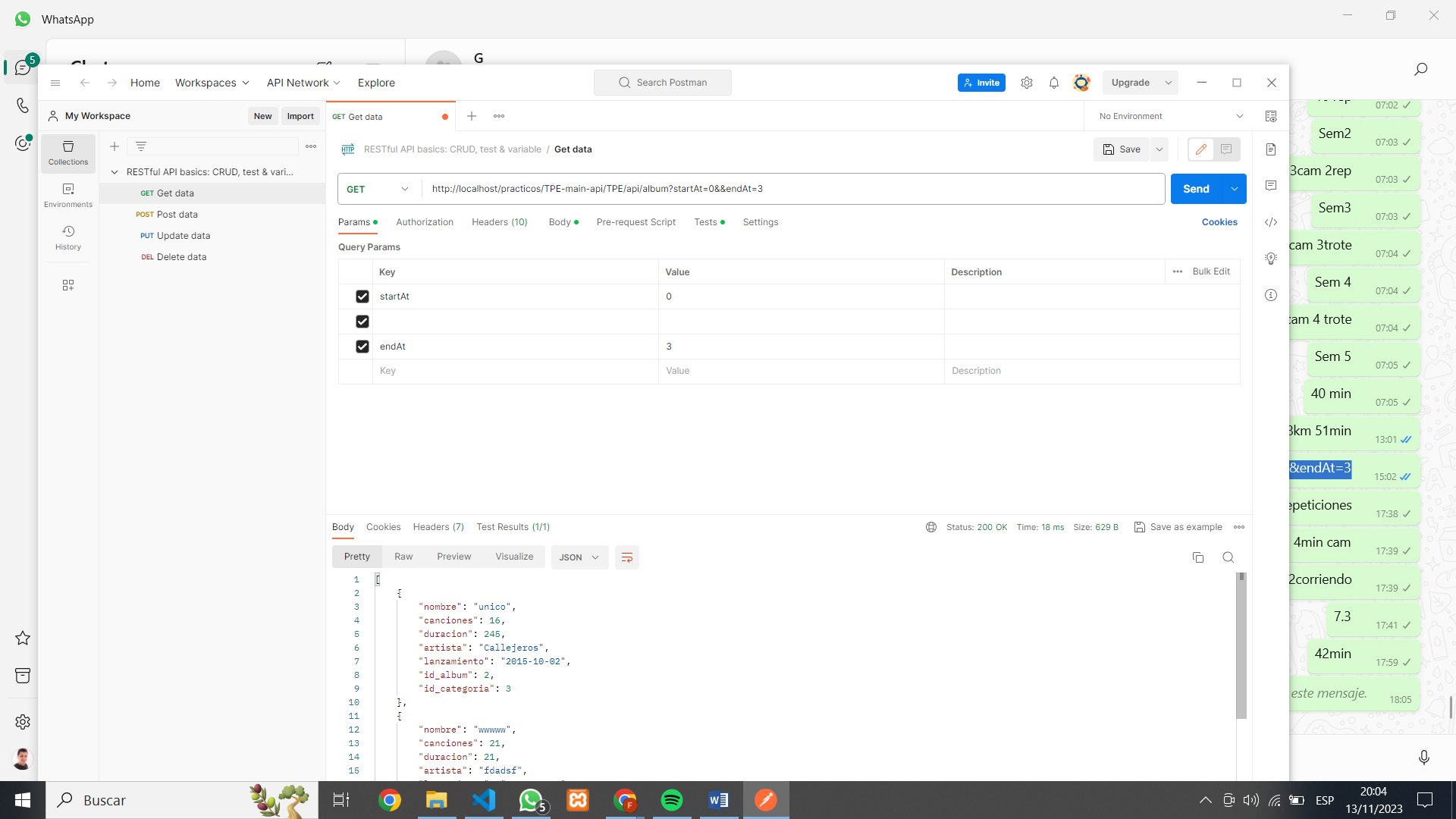
En el siguiente ejemplo ordeno de la “a” a la “z” el nombre de los álbumes.



En caso de error de url devolvemos error 404.

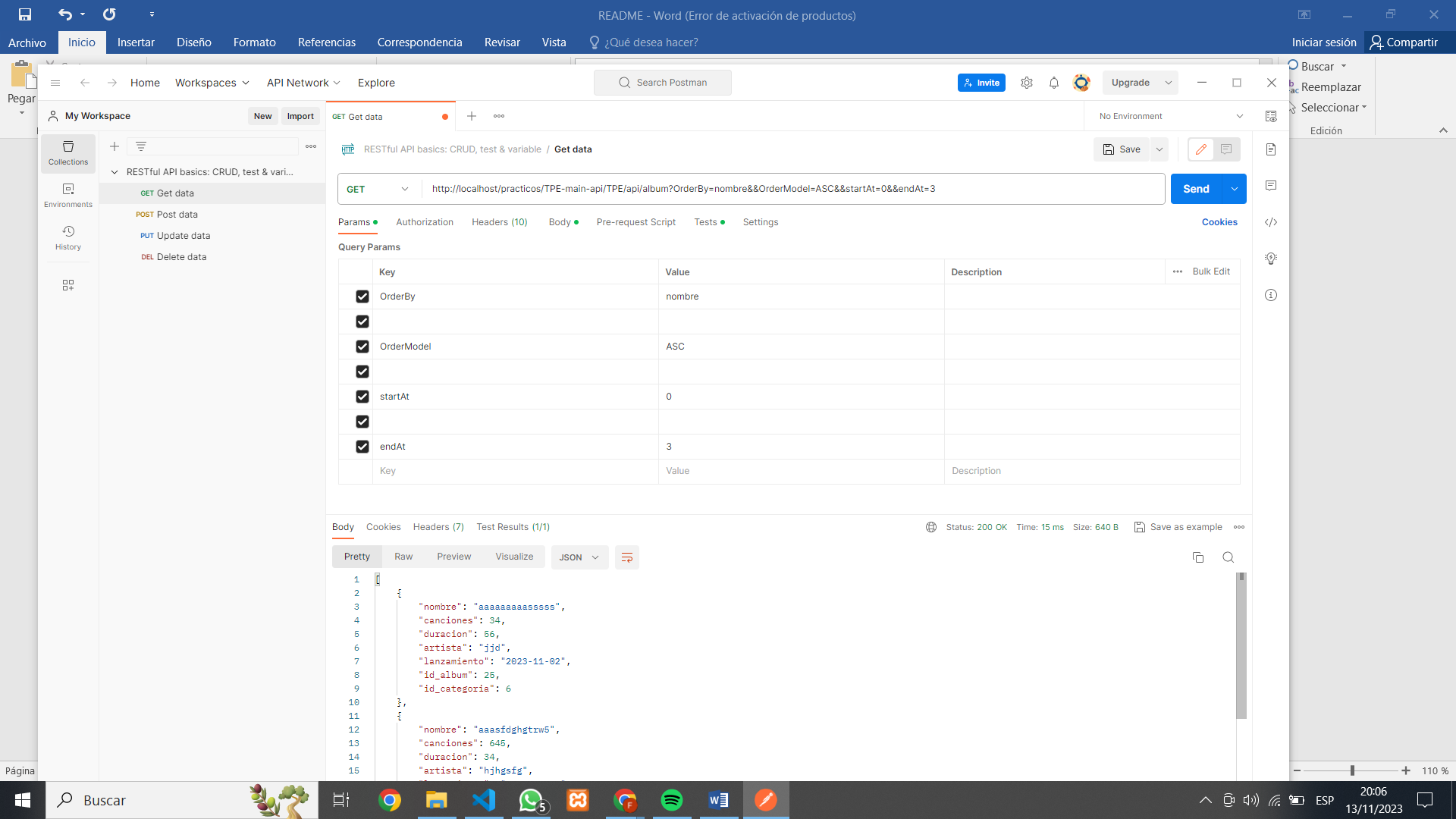
**Paginado,** agregando los parámetros startAt (indicar la posición inicial desde donde se empiezan a limitar, la primera posición es el 0) y endAt (indica la cantidad de datos que se desean paginar).

El siguiente ejemplo indica que me van a mostrar solo los primeros 3 álbumes.



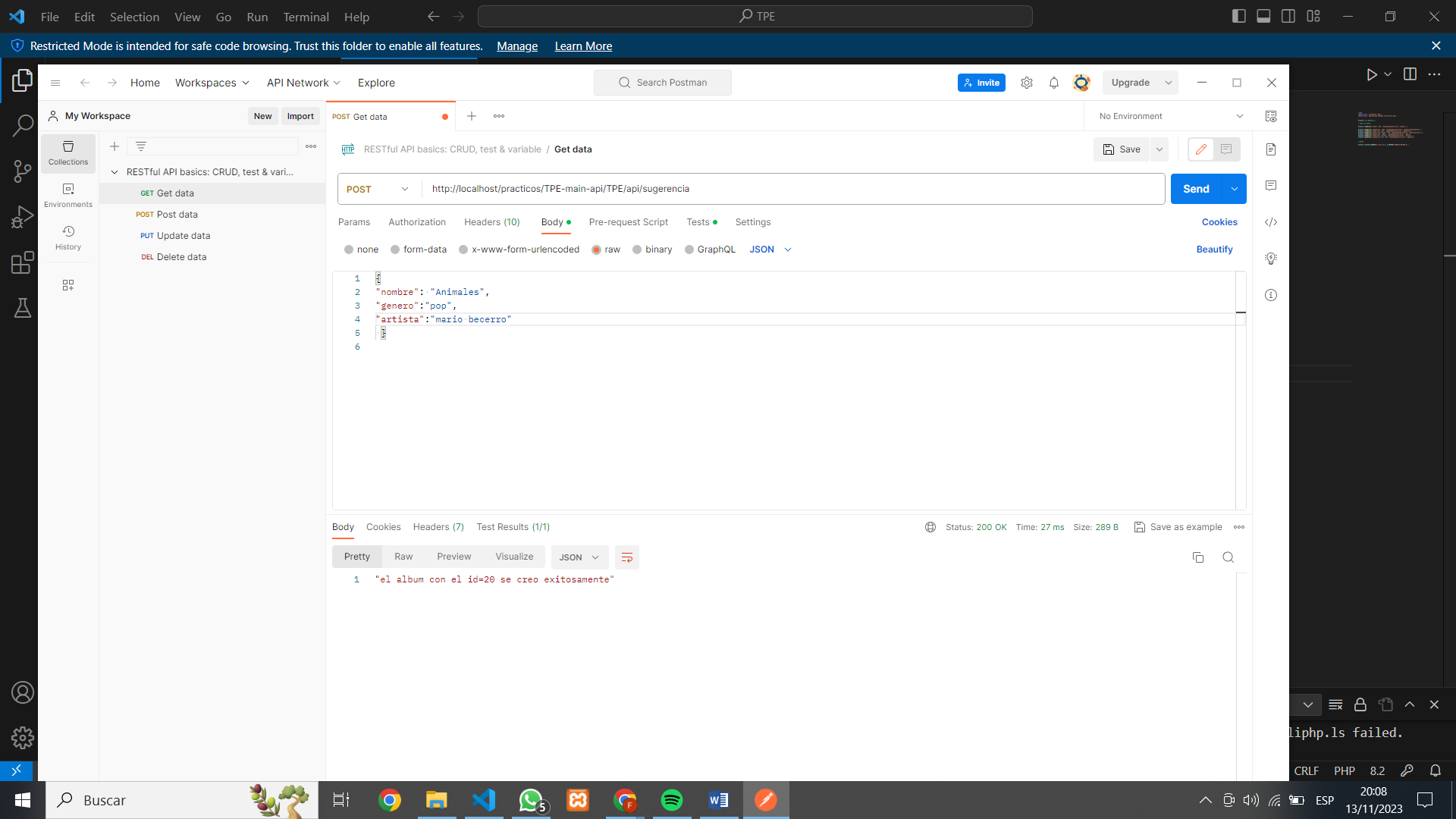
En caso de error de url devolvemos error 404.

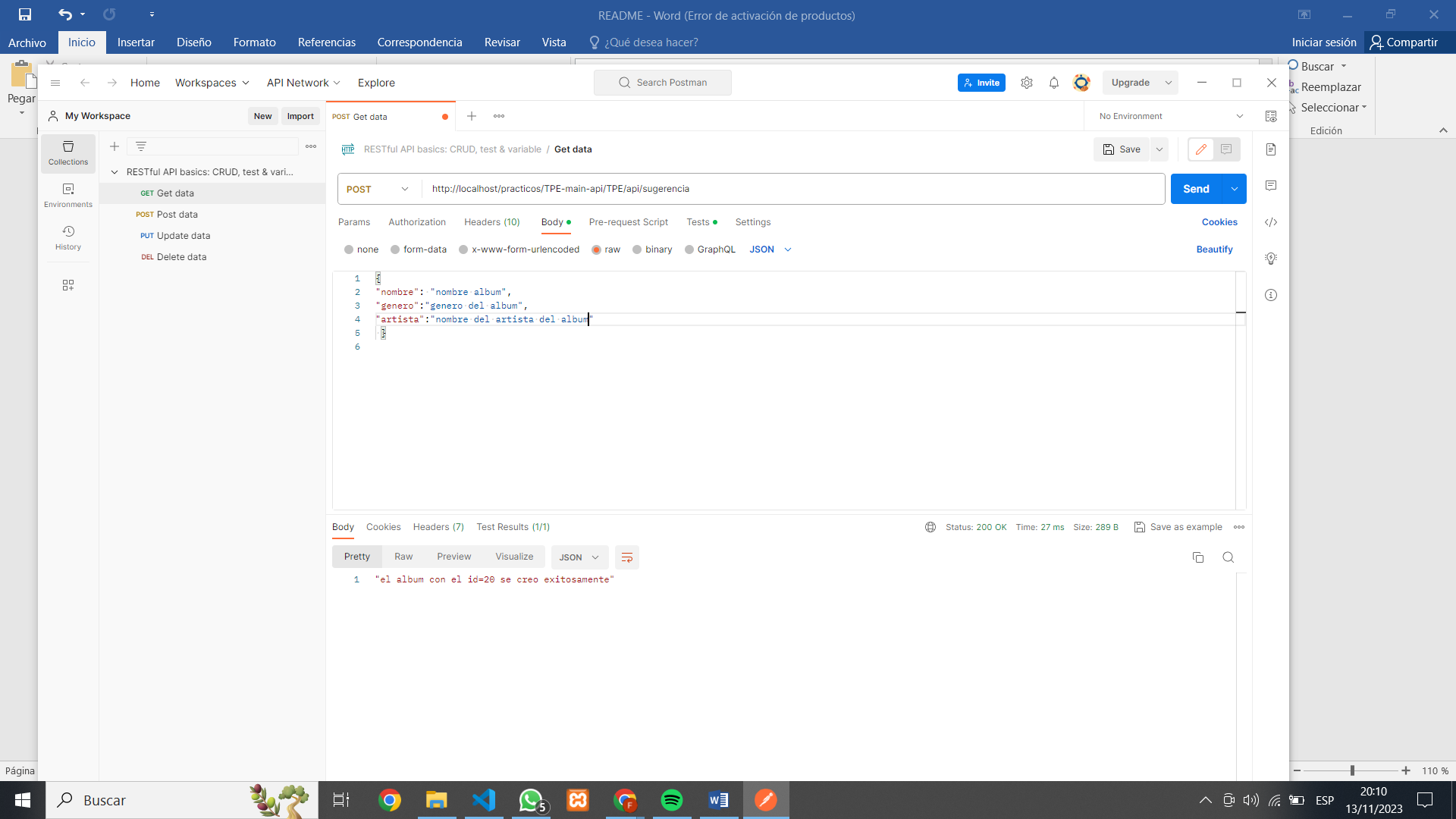
También nuestro api fue diseñado para que se puedan ordenar y paginar a la vez, como lo muestro en el siguiente ejemplo.



**Solicitudes de tipo POST:**

**Insertar álbum sugerido**, esta opción permite que los usuarios puedan dejar de forma sugerida álbumes, pueden agregar (nombre del álbum, género y artista).

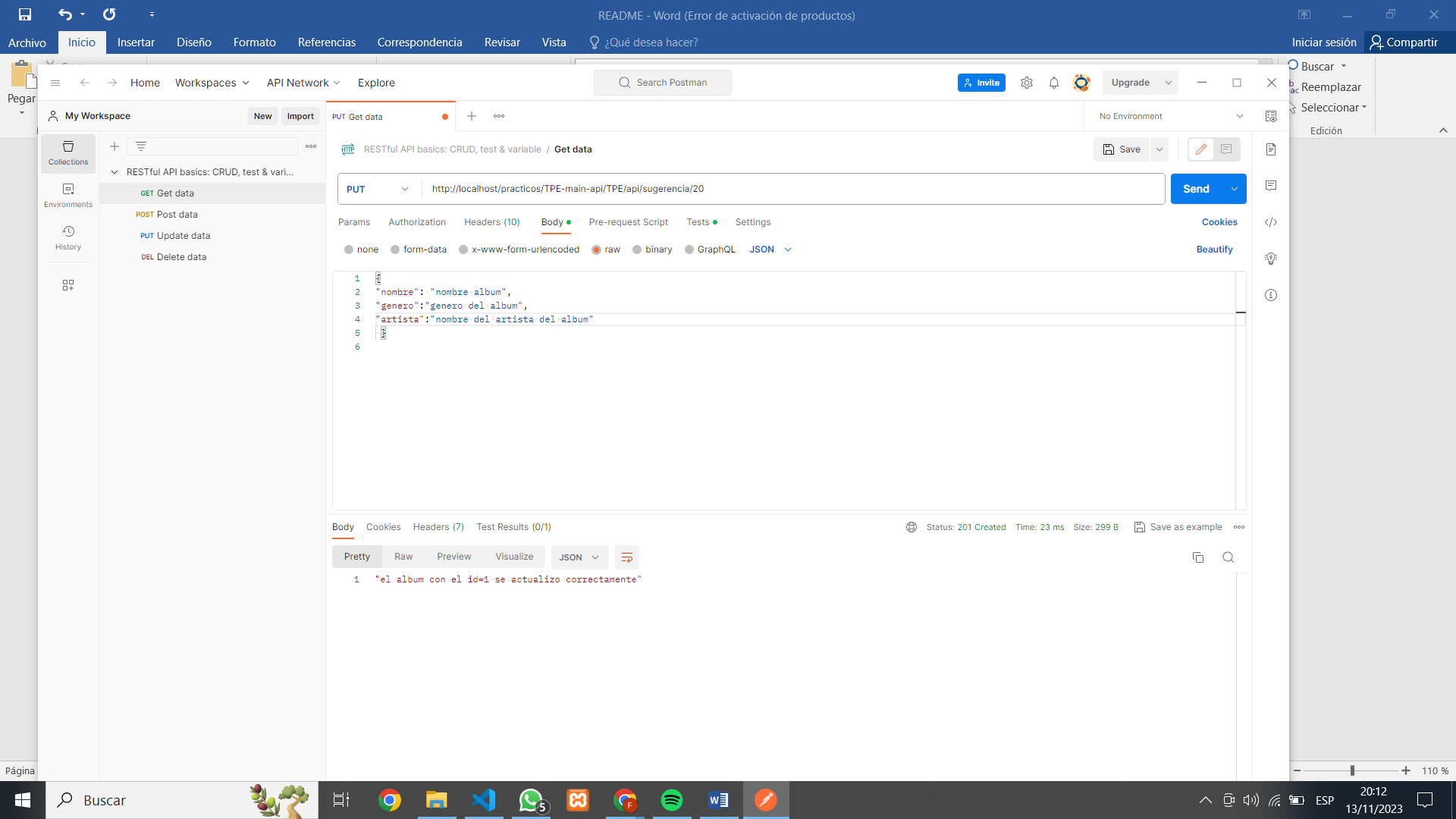




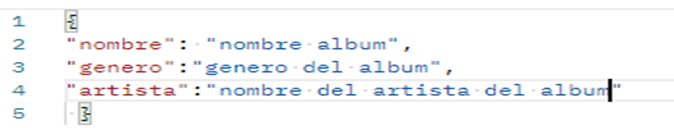
Para hacer un post todos los datos deben estar cargados de lo contrario dará un recurso de tipo 404.

**Solicitudes de tipo PUT:**

**Editar datos del álbum sugeridos por el usuario**, para editar los datos sugeridos por el usuario.



Es fundamental saber el id del álbum a modificar, una vez tengamos definido el id del álbum es obligatoria seguir la misma sintaxis del post, es decir:



Si ingresa datos vacíos, devolverá un recurso de tipo 404.