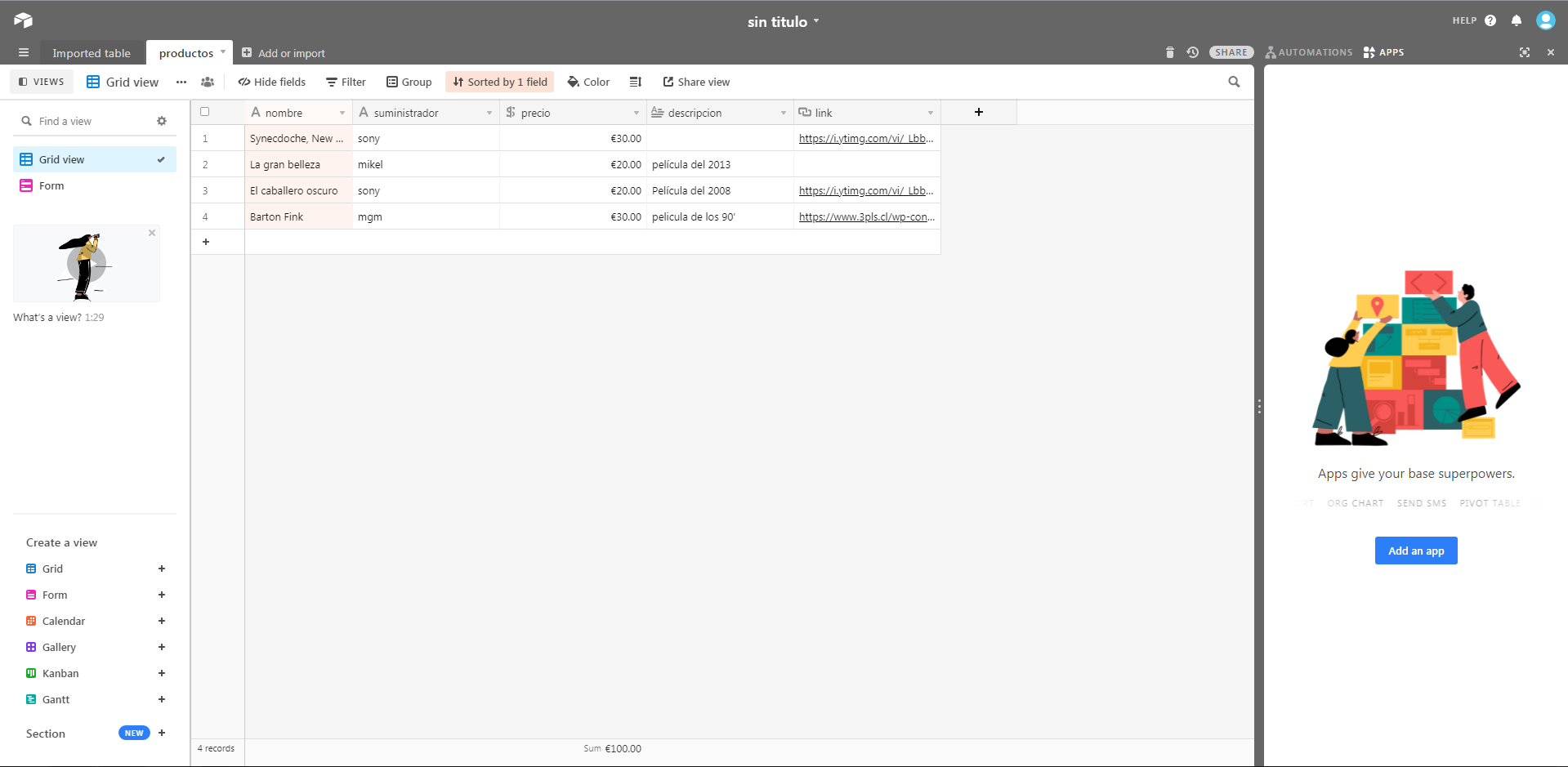
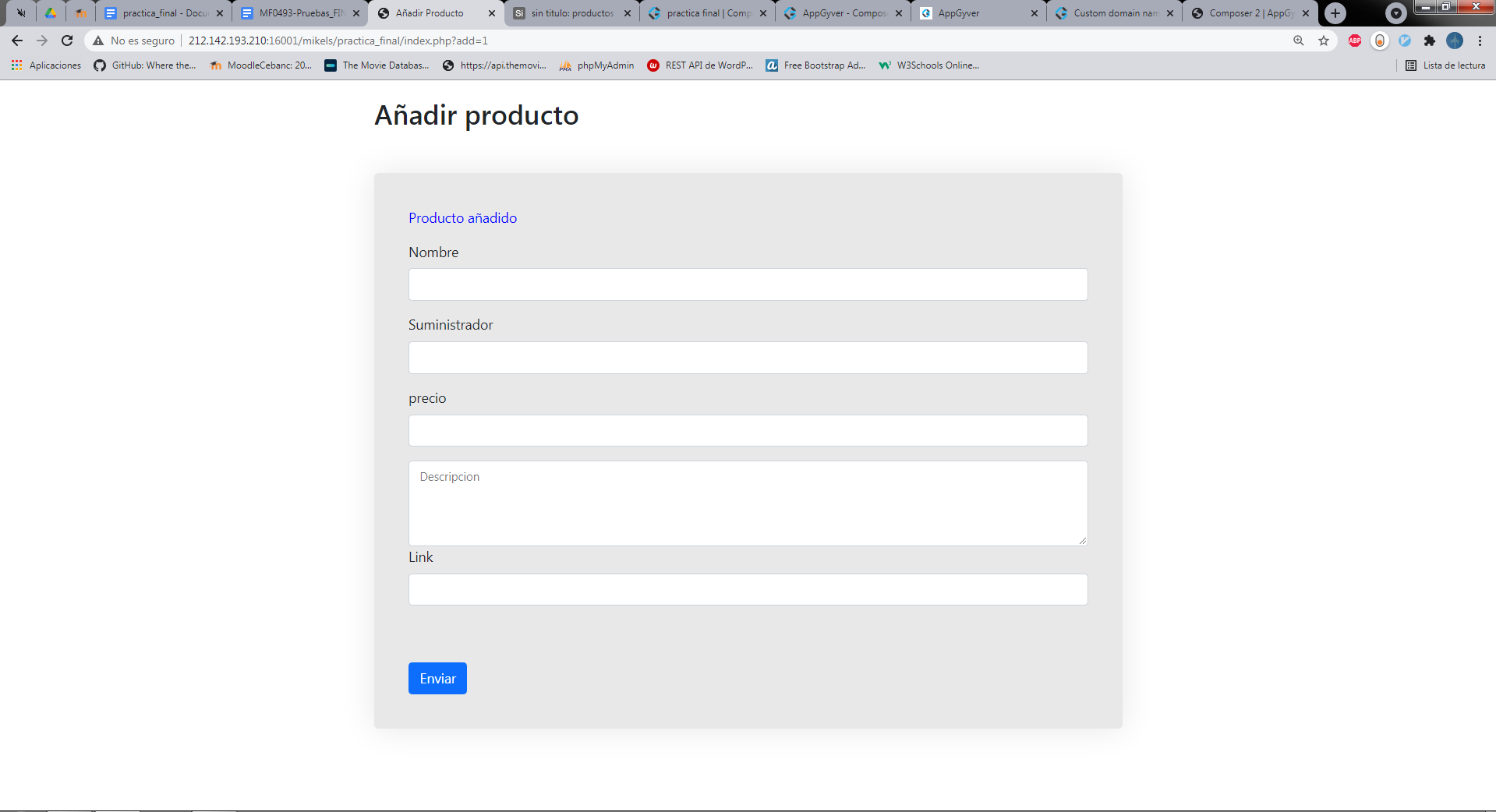
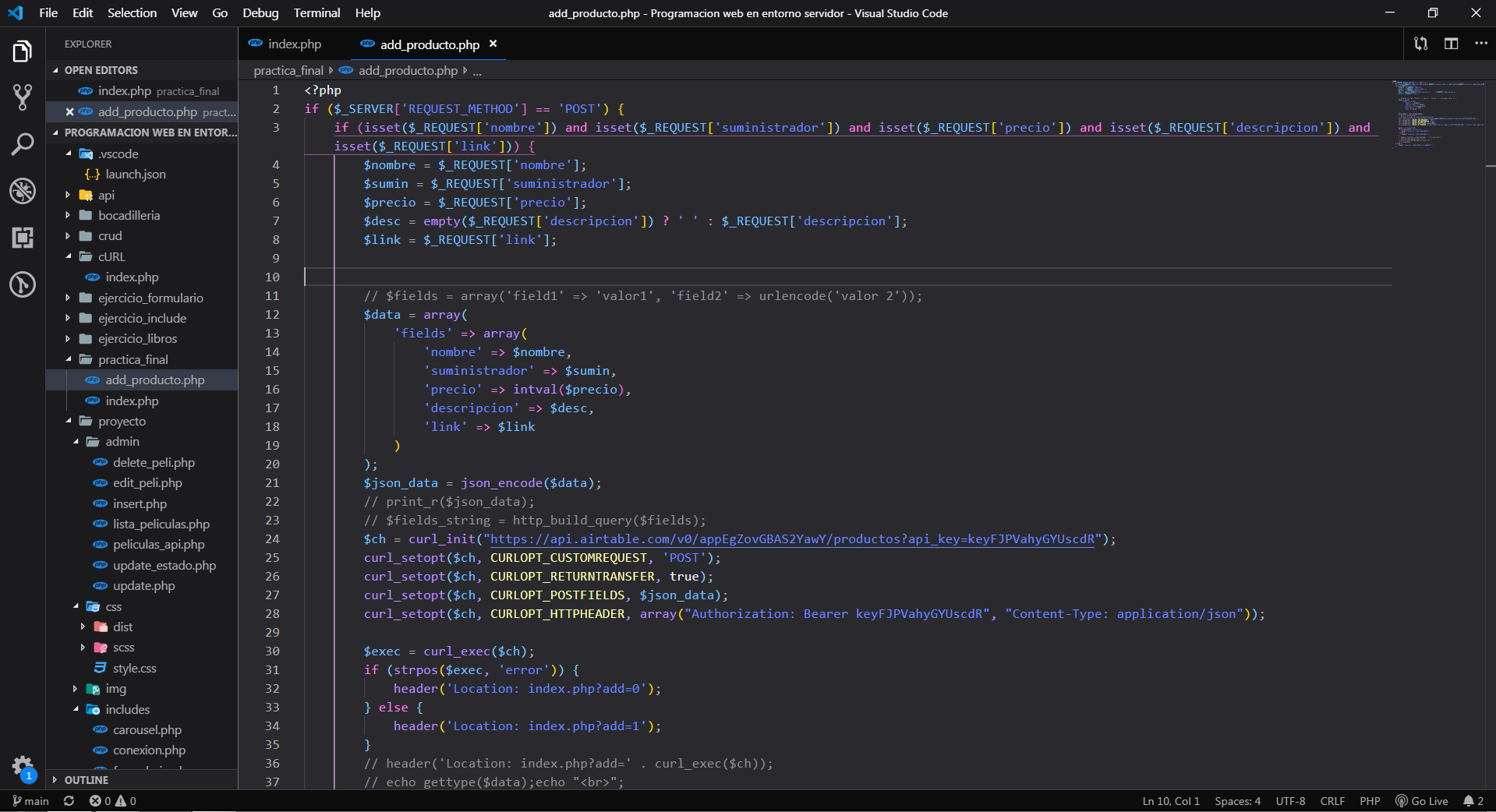
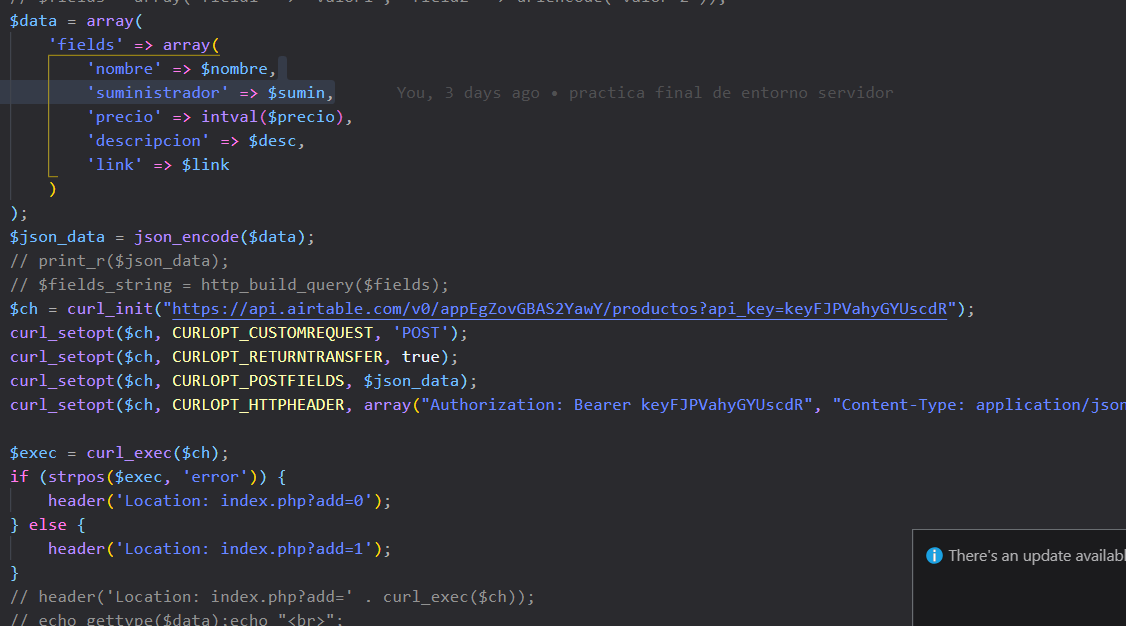
Creamos la tabla productos en airtable

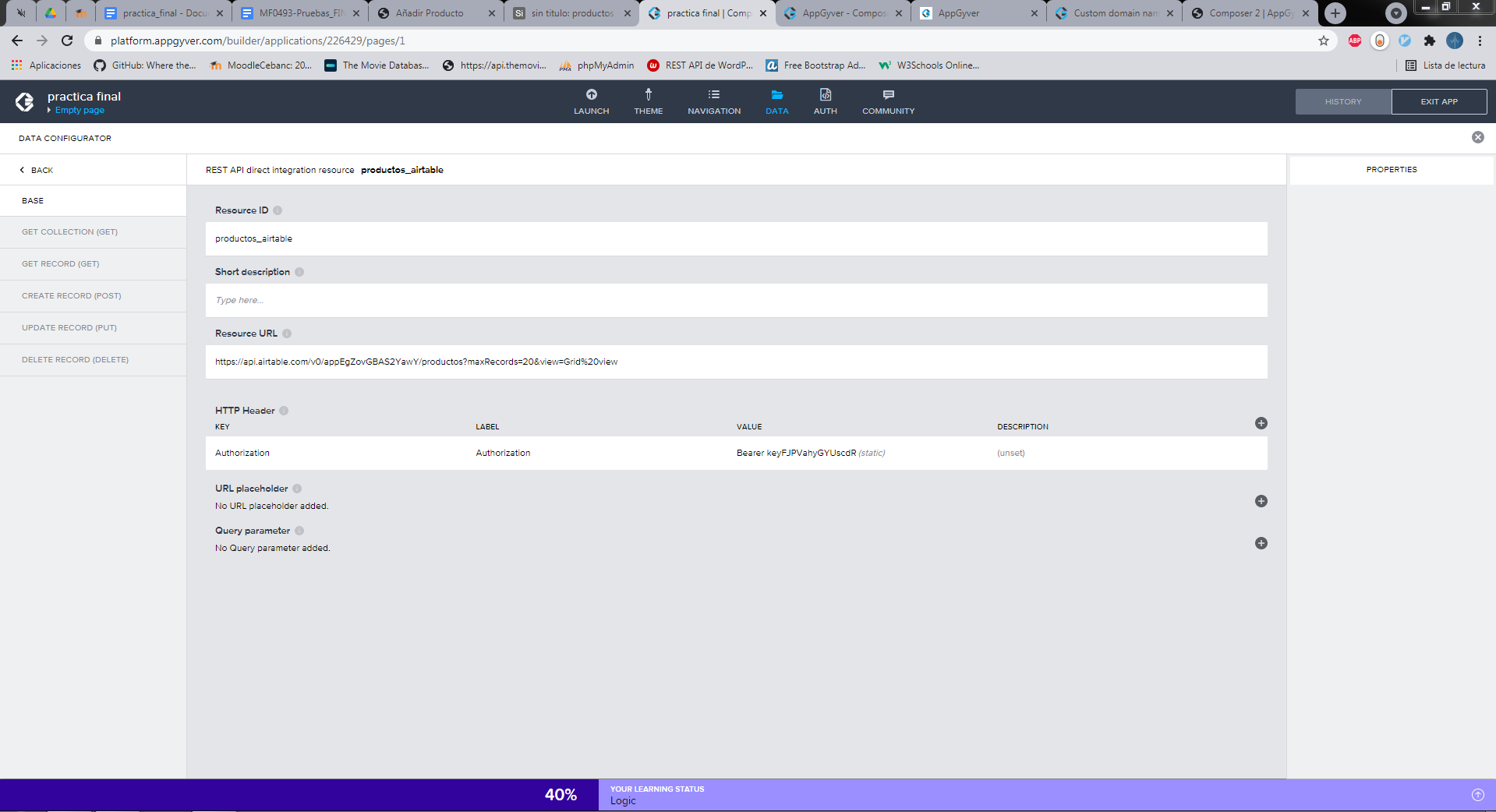
Creamos el formulario para añadir productos con php

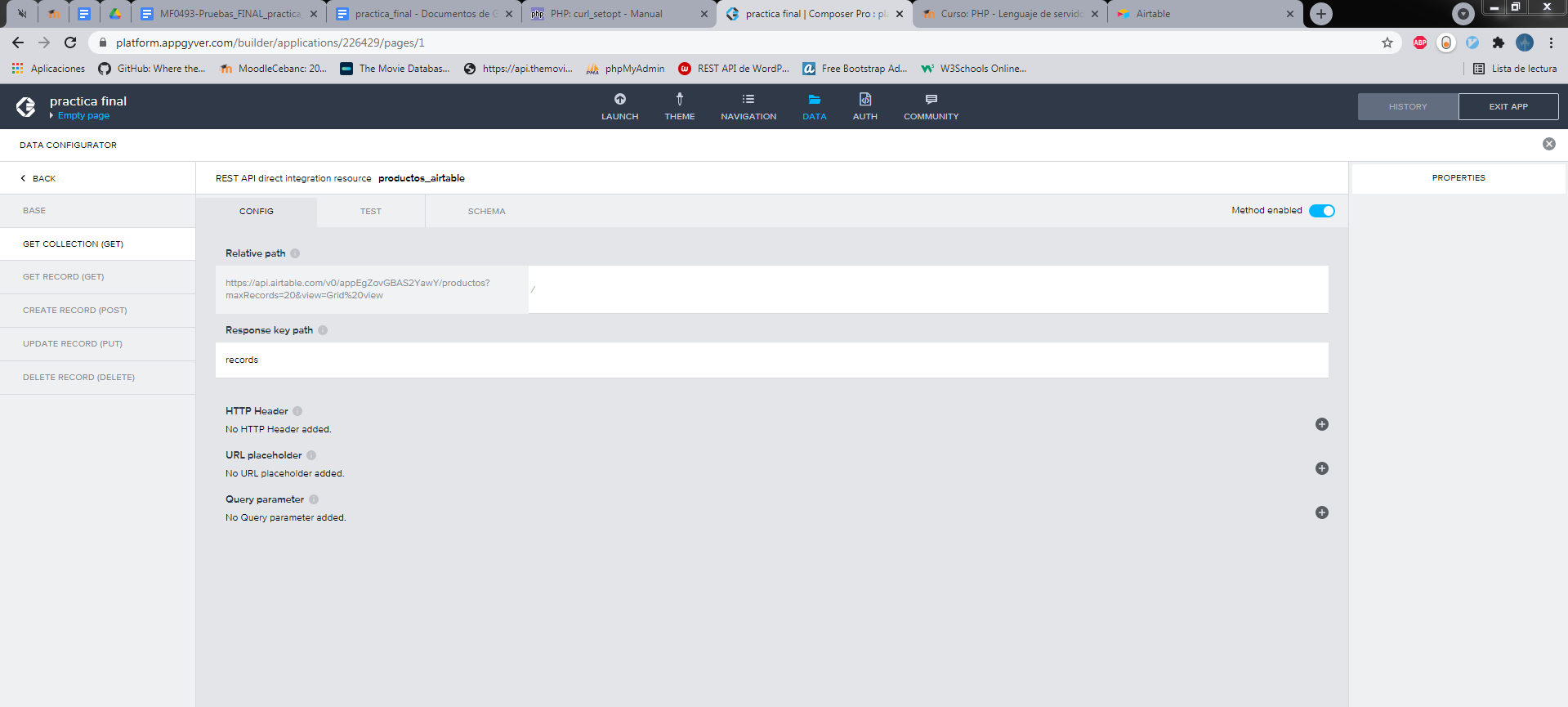
La inserción la hacemos mediante la api de airtable utilizando cURL



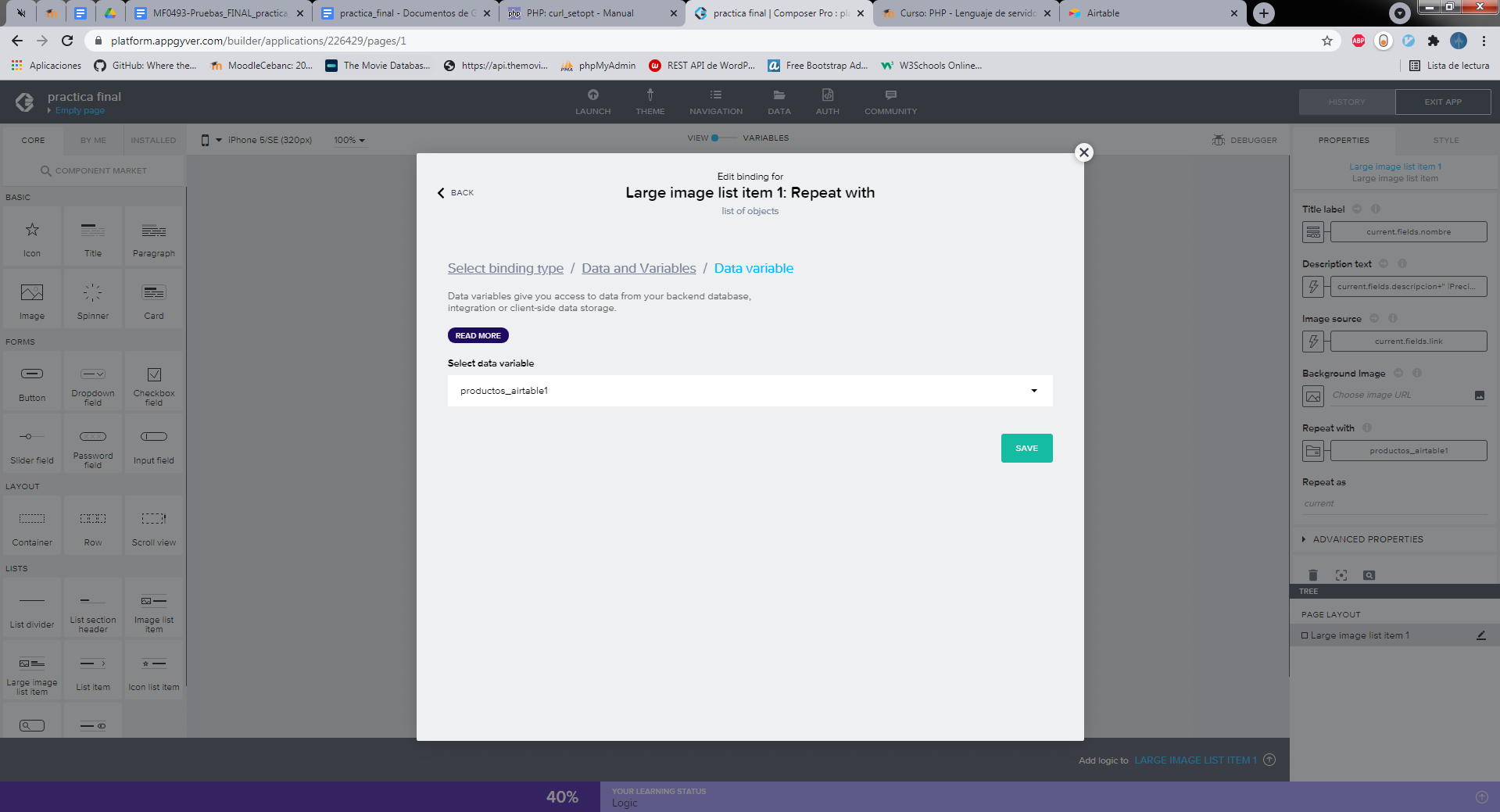
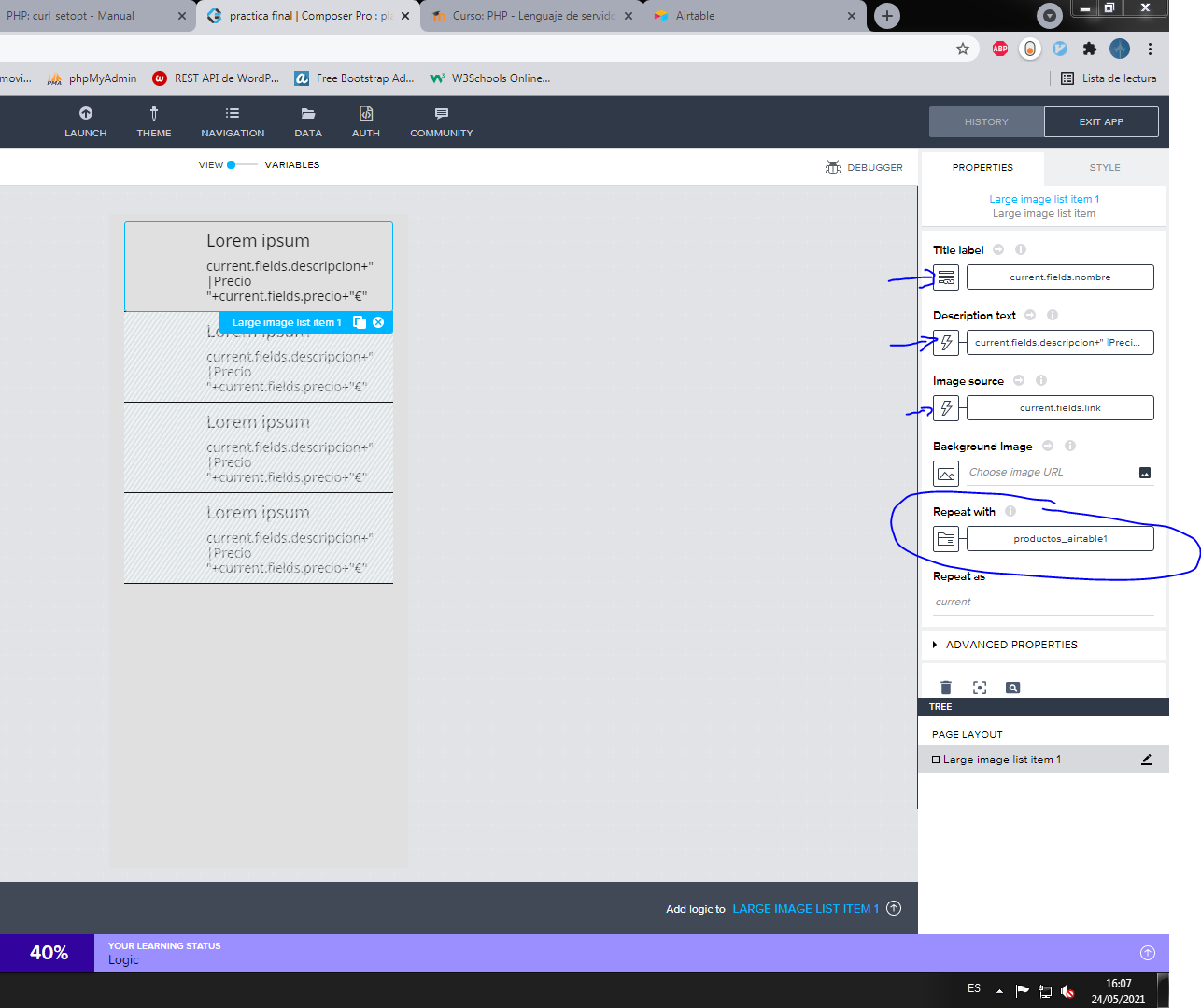
Introducimos la url de la api con la clave que nos da airtable en el método curl\_init().

Después configuramos la sesión de cURL: ponemos el método de petición en POST, después le damos valor true a la opción de CURLOP\_RETURNTRANSFER para que la respuesta que nos devuelva la api sea un string y así podamos detectar mediante el método ‘strpos()’ la palabra ‘error’ en el string devuelto. Pasamos los datos del formulario con CURLOPT\_POSTFIELDS, y mediante CURLOPT\_HTTPHEADER mandamos también la variable ‘Authorization’ que nos dá airtable y la variable content-type. Para terminar lo ejecutamos y si encontramos la palabra ‘error’ en el json devuelto informamos de que algo ha ido mal.

Para mostrar una lista de los productos en una aplicación creada en AppGyver, creamos primero la aplicación. A continuación vamos a la pestaña ‘Data’ y añadimos una conexión a una api, en nuestro caso será la de airtable. Introducimos la url de la api y también añadimos la variable authorization con la clave que nos da airtable.

En la pestaña de ‘Get Collection’ en el campo ‘response key path’ le decimos que empiece por ‘records’ porque el primer elemento que nos da airtable es ‘records’ (registros) de una tabla, y lo que nos interesa es que empiece por el elemento registros para listarlos.

En la parte que está redondeada le añadimos la variable que hemos creado anteriormente que está asociada a la api y que recoge los registros de productos



Por último en los campos de title-table, description-text y image-source le damos los valores de título, descripción y ruta de la imagen del actual producto.

URL de la web🡪<https://mikel-practica-final.appgyverapp.com/>