



Universidad Gerardo Barrios

Facultad de Ciencia y Tecnología

Ingeniería en Sistemas y Redes Informáticas

Estudiante:

Vasti Gabriela Flores Zelaya

Asignatura:

Programación Computacional IV

Docente:

Inga. Gisela Yasmín Espinoza

Fecha de entrega:

16 de mayo, 2021

USANDO ASINCRONISMO EN NODEJS, CONSULTA API'S

Consulta a valor del bitcoin mediante la API

```
api.coindesk.com/v1/bpi/currentprice.json

{"time":{"updated":"May 14, 2021 02:45:00 UTC","updatedISO":"2021-05-14T02:45:00+00:00","updateduk":"May 14, 2021 at 03:45 BST"},"disclaimer":"This data was produced from the CoinDesk Bitcoin Price Index (USD). Non-USD currency data converted using hourly conversion rate from openexchangerates.org","chartName":"Bitcoin","bpi":{"USD":{"code":"USD","symbol":"$","rate":"49,734.8895","description":"United States Dollar","rate_float":49734.8895},"GBP":{"code":"GBP","symbol":"£","rate":"35,408.5556","description":"British Pound Sterling","rate_float":35408.5556},"EUR":{"code":"EUR","symbol":"€","rate":"41,162.4845","description":"Euro","rate_float":41162.4845}}}}
```

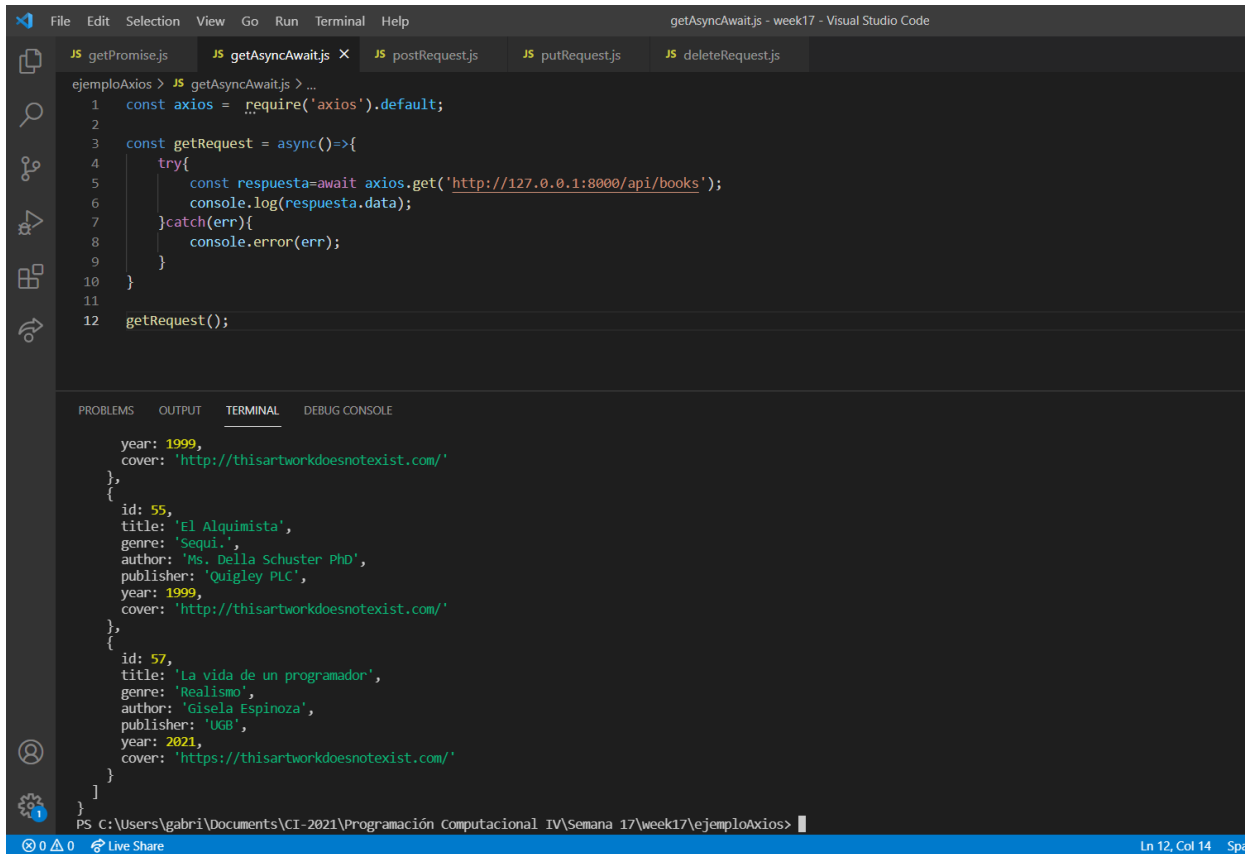
```
apiBitcoin.js
1 const request=require('request');
2
3 request('https://api.coindesk.com/v1/bpi/currentprice.json', (error, response, bitcoin) => {
4   if(error){
5     console.error('could not send request to API: ${error.message}');
6     return;
7   }
8   if (response.statusCode != 200) {
9     console.error('Expected status code 200 but received ${response.statusCode}.');
10    return;
11  }
12  console.log('Processing our price of bitcoin');
13  console.log('El precio del bitcoin se detalla a continuación: \n${bitcoin}');
14
15 })
16
17
```

```
PS C:\Users\gabri\Documents\CI-2021\Programación Computacional IV\Semana 17\week17> node apiBitcoin
bitcoin
El precio del bitcoin se detalla a continuación:
{"time":{"updated":"May 14, 2021 02:58:00 UTC","updatedISO":"2021-05-14T02:58:00+00:00","updateduk":"May 14, 2021 at 03:58 BST"},"disclaimer":"This data was produced from the coinDesk Bitcoin Price Index (USD). Non-USD currency data converted using hourly conversion rate from openexchangerates.org","chartName":"Bitcoin","bpi":{"USD":{"code":"USD","symbol":"$","rate":"49,590.6376","description":"United States Dollar","rate_float":49590.6376},"GBP":{"code":"GBP","symbol":"£","rate":"35,305.8561","description":"British Pound Sterling","rate_float":35305.8561},"EUR":{"code":"EUR","symbol":"€","rate":"41,043.0961","description":"Euro","rate_float":41043.0961}}}}
```

CONSULTA A API LOCAL SOBRE LIBROS

GET

Lista registro de la API



The screenshot shows a Visual Studio Code editor with a file named `getAsyncAwait.js` open. The code in the file is as follows:

```
1 const axios = require('axios').default;
2
3 const getRequest = async()=>{
4   try{
5     const respuesta=await axios.get('http://127.0.0.1:8000/api/books');
6     console.log(respuesta.data);
7   }catch(err){
8     console.error(err);
9   }
10 }
11
12 getRequest();
```

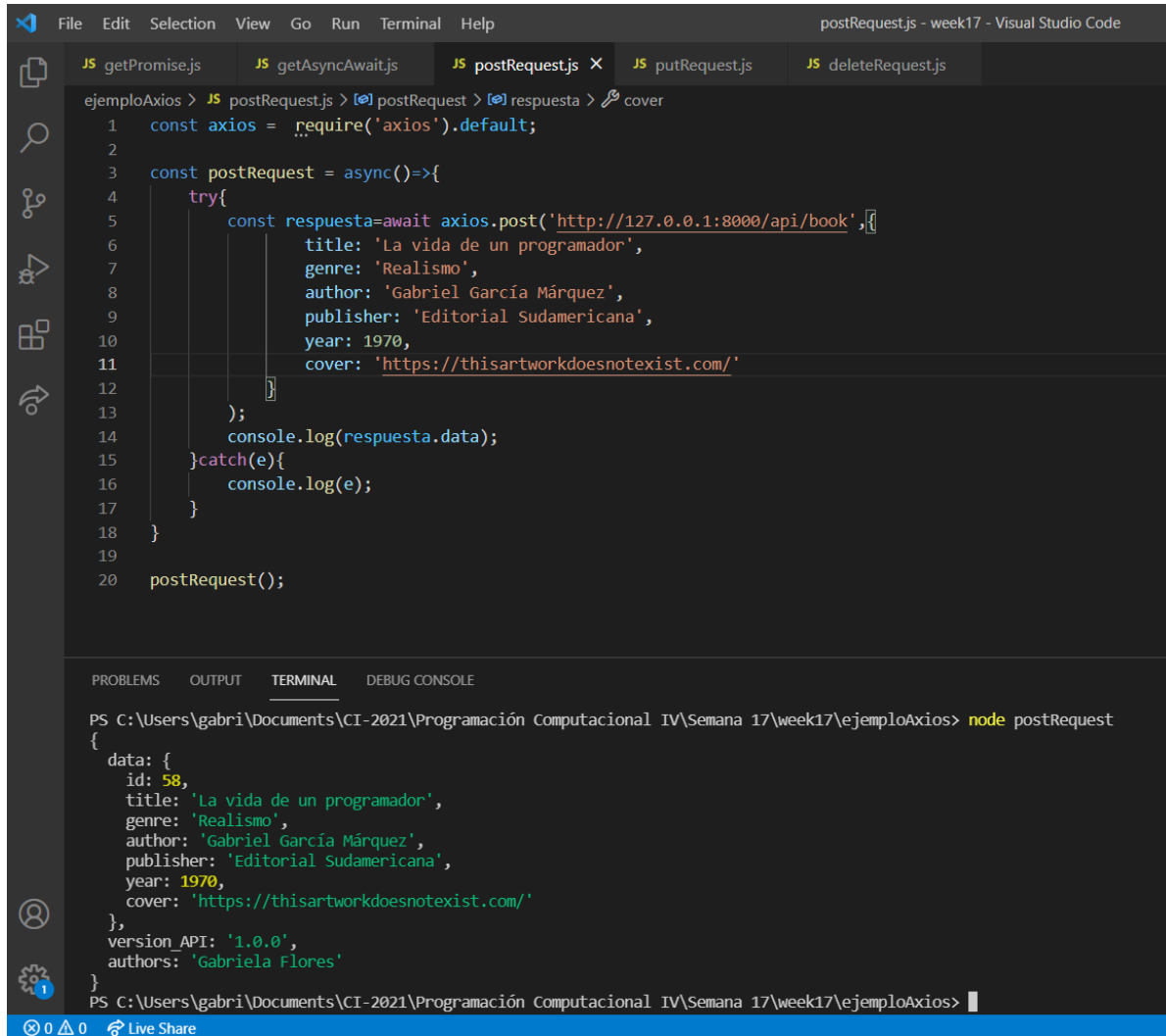
The terminal output at the bottom shows the response from the API:

```
year: 1999,
cover: 'http://thisartworkdoesnotexist.com/'
},
{
  id: 55,
  title: 'El Alquimista',
  genre: 'Sequi.',
  author: 'Ms. Della Schuster PhD',
  publisher: 'Quigley PLC',
  year: 1999,
  cover: 'http://thisartworkdoesnotexist.com/'
},
{
  id: 57,
  title: 'La vida de un programador',
  genre: 'Realismo',
  author: 'Gisela Espinoza',
  publisher: 'UGB',
  year: 2021,
  cover: 'https://thisartworkdoesnotexist.com/'
}
]
```

The status bar at the bottom indicates the file is at line 12, column 14.

POST

Agrega un registro



The screenshot shows the Visual Studio Code editor with a file named `postRequest.js` open. The code in the editor is as follows:

```
1 const axios = require('axios').default;
2
3 const postRequest = async()=>{
4   try{
5     const respuesta=await axios.post('http://127.0.0.1:8000/api/book',{
6       title: 'La vida de un programador',
7       genre: 'Realismo',
8       author: 'Gabriel García Márquez',
9       publisher: 'Editorial Sudamericana',
10      year: 1970,
11      cover: 'https://thisartworkdoesnotexist.com/'
12    });
13   }catch(e){
14     console.log(e);
15   }
16 }
17
18 postRequest();
```

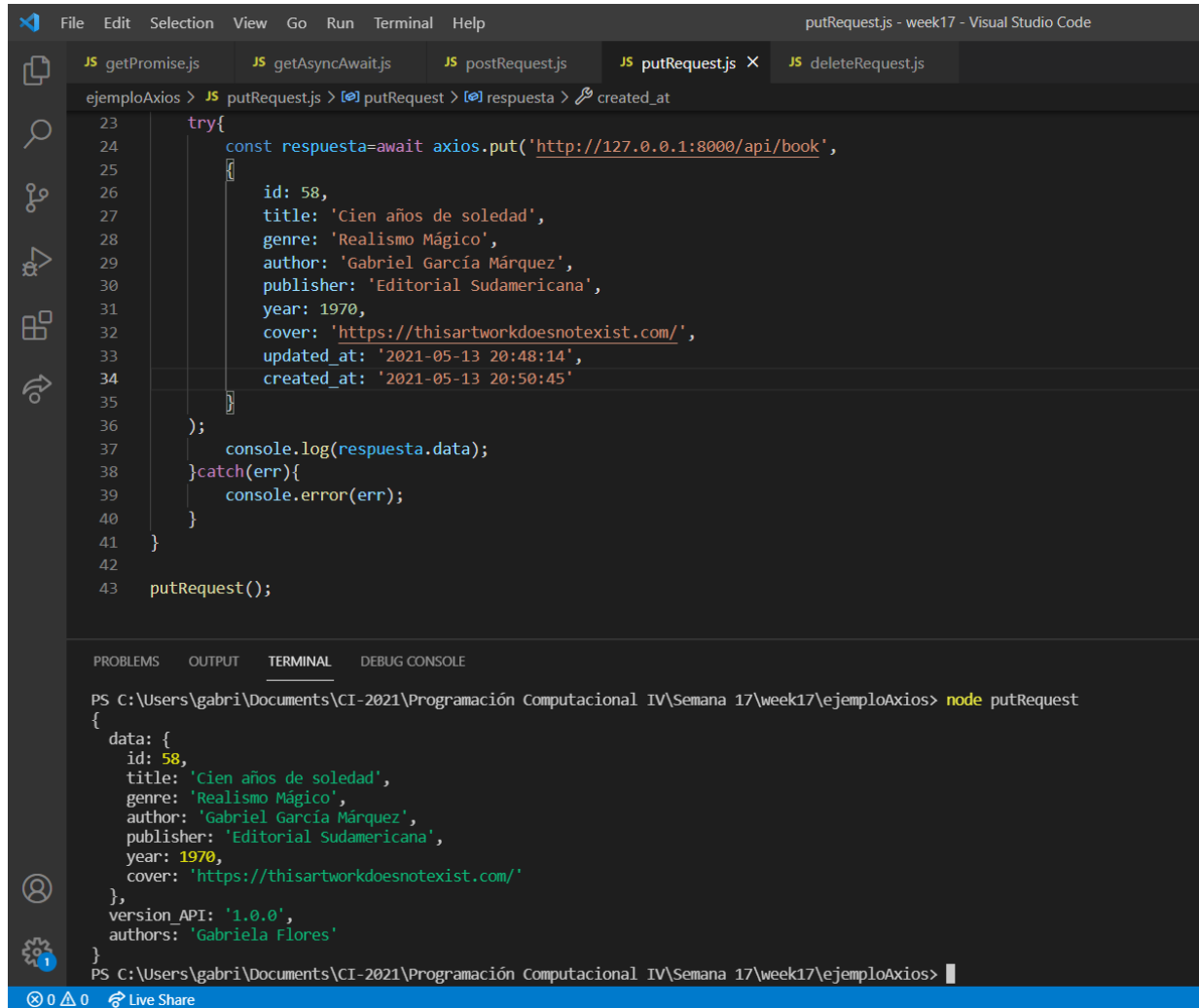
Below the editor, the TERMINAL panel shows the output of running `node postRequest` in a PowerShell prompt:

```
PS C:\Users\gabri\Documents\CI-2021\Programación Computacional IV\Semana 17\week17\ejemploAxios> node postRequest
{
  data: {
    id: 58,
    title: 'La vida de un programador',
    genre: 'Realismo',
    author: 'Gabriel García Márquez',
    publisher: 'Editorial Sudamericana',
    year: 1970,
    cover: 'https://thisartworkdoesnotexist.com/'
  },
  version_API: '1.0.0',
  authors: 'Gabriela Flores'
}
```

The status bar at the bottom indicates 0 errors, 0 warnings, and 0 info messages, along with a Live Share icon.

PUT

Actualiza campos del registro con id 58



The image shows a Visual Studio Code editor window with a file named `putRequest.js` open. The code is a JavaScript file using `axios` to send a PUT request to a local API. The request body contains book details for a book with ID 58, including title, genre, author, publisher, year, cover, and timestamps for `updated_at` and `created_at`. The code uses `try/catch` to handle the response or any errors.

```
23 try{
24   const respuesta=await axios.put('http://127.0.0.1:8000/api/book',
25     {
26       id: 58,
27       title: 'Cien años de soledad',
28       genre: 'Realismo Mágico',
29       author: 'Gabriel García Márquez',
30       publisher: 'Editorial Sudamericana',
31       year: 1970,
32       cover: 'https://thisartworkdoesnotexist.com/',
33       updated_at: '2021-05-13 20:48:14',
34       created_at: '2021-05-13 20:50:45'
35     }
36   );
37   console.log(respuesta.data);
38 }catch(err){
39   console.error(err);
40 }
41 }
42
43 putRequest();
```

Below the code editor, the `TERMINAL` tab is active, showing the command `node putRequest` and its output. The output is a JSON object representing the response from the API, which includes the book data and API version/authors.

```
PS C:\Users\gabri\Documents\CI-2021\Programación Computacional IV\Semana 17\week17\ejemploAxios> node putRequest
{
  data: {
    id: 58,
    title: 'Cien años de soledad',
    genre: 'Realismo Mágico',
    author: 'Gabriel García Márquez',
    publisher: 'Editorial Sudamericana',
    year: 1970,
    cover: 'https://thisartworkdoesnotexist.com/'
  },
  version_API: '1.0.0',
  authors: 'Gabriela Flores'
}
```

The status bar at the bottom shows the file encoding as `UTF-8` and the presence of a `Live Share` session.

DELETE

Consulta registro de archivos

```
PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

    year: 1999,
    cover: 'http://thisartworkdoesnotexist.com/'
  },
  {
    id: 57,
    title: 'La vida de un programador',
    genre: 'Realismo',
    author: 'Gisela Espinoza',
    publisher: 'UGB',
    year: 2021,
    cover: 'https://thisartworkdoesnotexist.com/'
  },
  {
    id: 58,
    title: 'Cien años de soledad',
    genre: 'Realismo Mágico',
    author: 'Gabriel García Márquez',
    publisher: 'Editorial Sudamericana',
    year: 1970,
    cover: 'https://thisartworkdoesnotexist.com/'
  }
]
PS C:\Users\gabri\Documents\CI-2021\Programación Computacional IV\Semana 17\week17\ejemploAxios>
```

Elimina registro id 58

```
File Edit Selection View Go Run Terminal Help
deleteRequest.js - week17 - Visual Studio Code

JS getPromise.js JS getAsyncAwait.js JS postRequest.js JS putRequest.js JS deleteRequest.js X

ejemploAxios > JS deleteRequest.js > deleteRequest > respuesta
1  const axios = require('axios').default;
2
3  const deleteRequest = async()=>{
4    try{
5      const respuesta=await axios.delete('http://127.0.0.1:8000/api/book/58')
6    };
7    console.log(respuesta.data);
8  }catch(err){
9    console.error(err);
10 }
11 }
12
13 deleteRequest();

PROBLEMS  OUTPUT  TERMINAL  DEBUG CONSOLE

PS C:\Users\gabri\Documents\CI-2021\Programación Computacional IV\Semana 17\week17\ejemploAxios> node deleteRequest
{
  data: {
    id: 58,
    title: 'Cien años de soledad',
    genre: 'Realismo Mágico',
    author: 'Gabriel García Márquez',
    publisher: 'Editorial Sudamericana',
    year: 1970,
    cover: 'https://thisartworkdoesnotexist.com/'
  },
  version_API: '1.0.0',
  authors: 'Gabriela Flores'
}
PS C:\Users\gabri\Documents\CI-2021\Programación Computacional IV\Semana 17\week17\ejemploAxios>
```

Consulta nuevamente registros, id 58 se ha eliminado

```
9 console.error(err);
10 }

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

id: 54,
title: 'El Alquimista',
genre: 'Sequi.',
author: 'Ms. Della Schuster PhD',
publisher: 'Quigley PLC',
year: 1999,
cover: 'http://thisartworkdoesnotexist.com/'
},
{
  id: 55,
  title: 'El Alquimista',
  genre: 'Sequi.',
  author: 'Ms. Della Schuster PhD',
  publisher: 'Quigley PLC',
  year: 1999,
  cover: 'http://thisartworkdoesnotexist.com/'
},
{
  id: 57,
  title: 'La vida de un programador',
  genre: 'Realismo',
  author: 'Gisela Espinoza',
  publisher: 'UGB',
  year: 2021,
  cover: 'https://thisartworkdoesnotexist.com/'
}
]
}
}

PS C:\Users\gabri\Documents\CI-2021\Programación Computacional IV\Semana 17\week17\ejemploAxios>
```

CONCLUSIONES

La librería Axios me permite realizar las peticiones HTTP, es una librería muy práctica y considero que sencilla de utilizar. Los métodos o verbos de HTTP, reciben dos parámetros; siendo el primero la url de la API y el segundo recibe el objeto en el cual personalizamos los datos para la petición. Pese a que es una librería relativamente fácil de utilizar no negaré que la asincronía sigue siendo un caso.

Sin duda hacer uso de las herramientas que ya existen puede agilizar el desarrollo y ahorrar tiempo, sin embargo aprender su trasfondo no dejara de ser siempre esencial e importante.