

Vue Fundamentals – 06

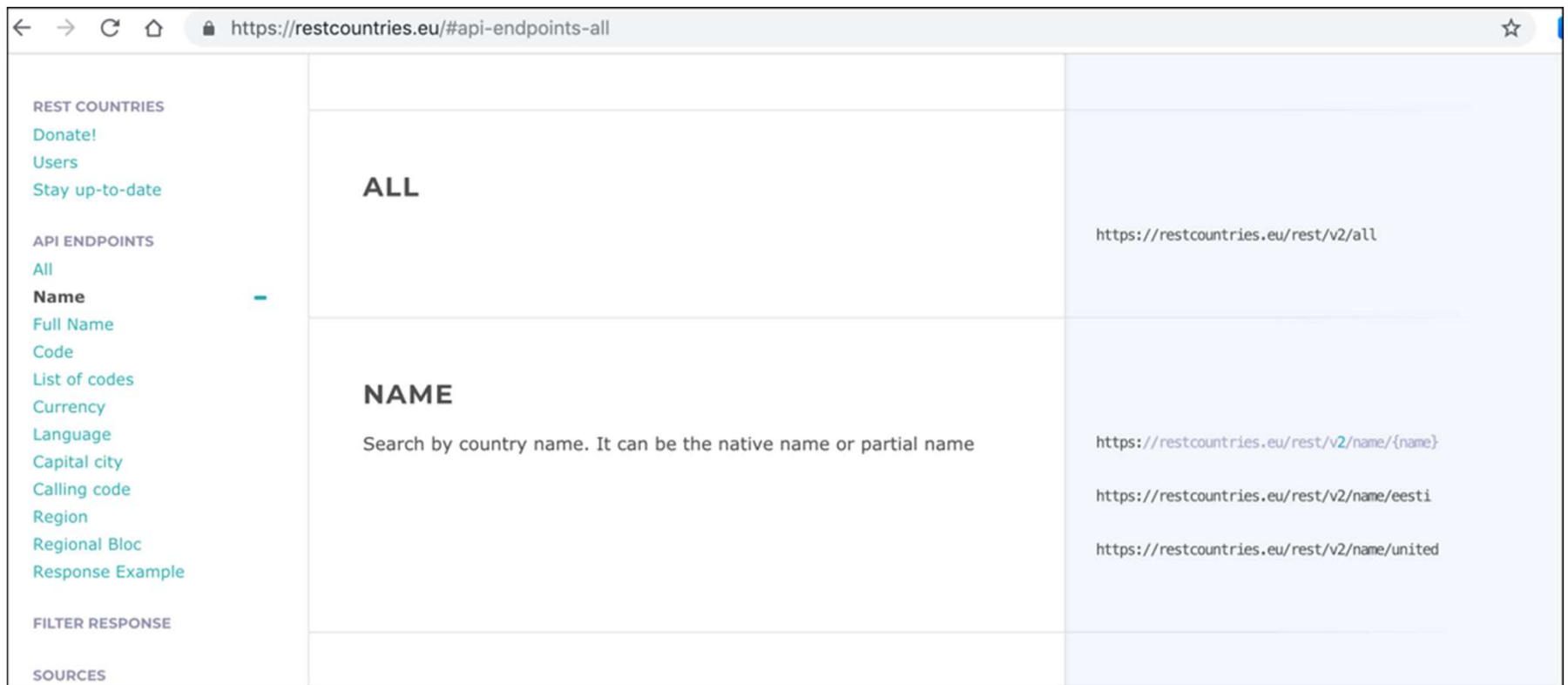
Using Http directly



Peter Kassenaar –
info@kassenaar.com

Consuming data from an API

- We're going to talk to a 'real' country-API on the internet
 - <https://restcountries.com/>
- Using `http` is *not* built into Vue.js



The screenshot shows the REST Countries API documentation website. The browser address bar displays `https://restcountries.eu/#api-endpoints-all`. The page is divided into three main sections: a left sidebar, a central content area, and a right sidebar.

Section	Item	Endpoint
API ENDPOINTS	All	<code>https://restcountries.eu/rest/v2/all</code>
	Name	<code>https://restcountries.eu/rest/v2/name/{name}</code>
FILTER RESPONSE	Full Name	<code>https://restcountries.eu/rest/v2/name/{name}</code>
	Code	<code>https://restcountries.eu/rest/v2/name/eeesti</code>
	List of codes	<code>https://restcountries.eu/rest/v2/name/united</code>

The left sidebar contains links for REST COUNTRIES (Donate!, Users, Stay up-to-date), API ENDPOINTS (All, Name, Full Name, Code, List of codes, Currency, Language, Capital city, Calling code, Region, Regional Bloc, Response Example), FILTER RESPONSE, and SOURCES.

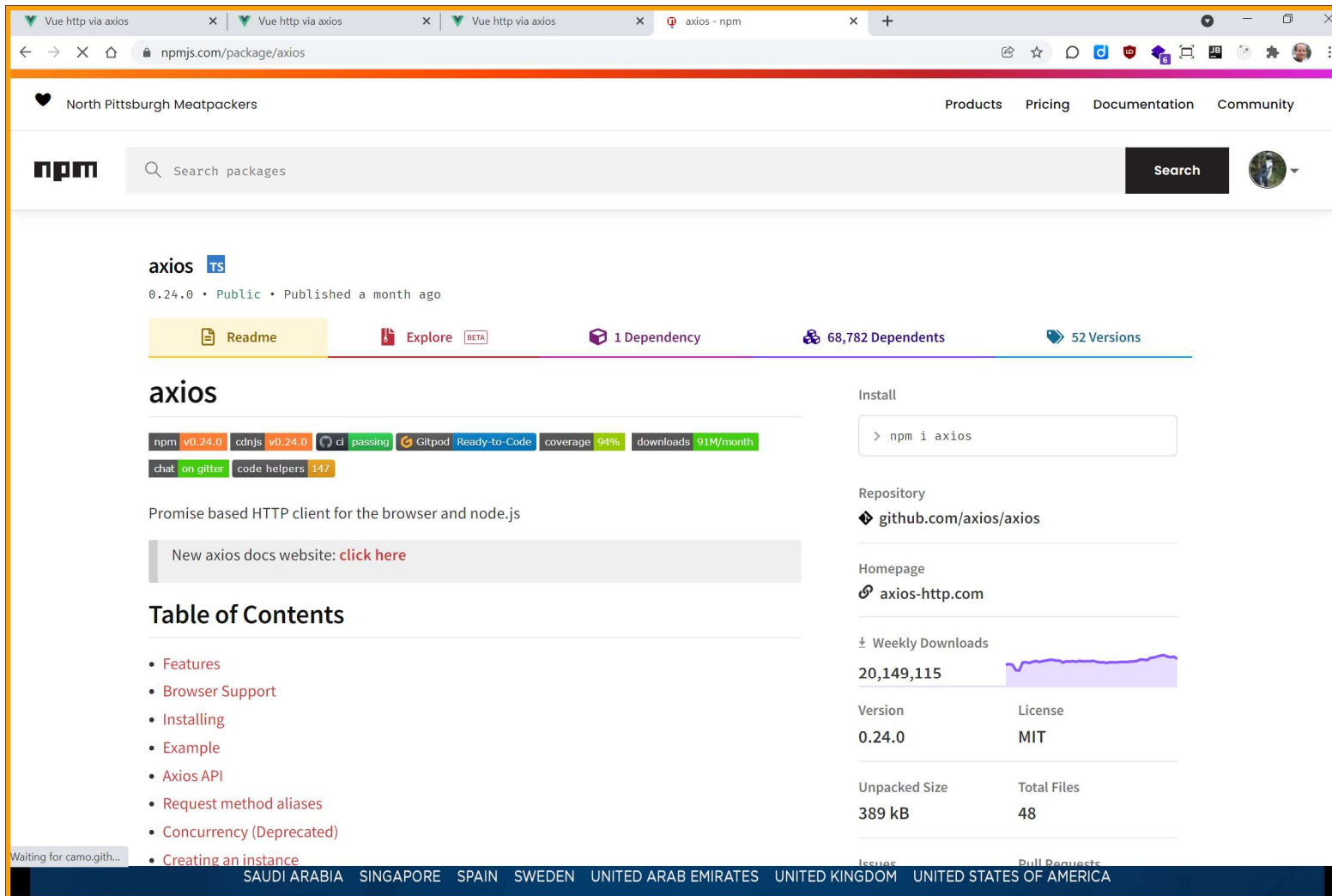
Using Axios

*"There are many times when building application for the web that you may want to consume and display data from an API. There are several ways to do so, but a very popular approach is to use **axios**, a promise-based HTTP client."*

<https://github.com/axios/axios>

<https://vuejs.org/v2/cookbook/using-axios-to-consume-apis.html>

<https://www.npmjs.com/package/axios>



The screenshot shows the npm package page for 'axios'. The browser's address bar displays 'npmjs.com/package/axios'. The page header includes the npm logo, a search bar, and navigation links for 'Products', 'Pricing', 'Documentation', and 'Community'. The package name 'axios' is prominently displayed with a TypeScript (TS) icon, followed by its version '0.24.0', 'Public' status, and 'Published a month ago'. Below this, there are links for 'Readme', 'Explore', '1 Dependency', '68,782 Dependents', and '52 Versions'. A horizontal bar contains various badges: 'npm v0.24.0', 'cdnjs v0.24.0', 'ci passing', 'Gitpod Ready-to-Code', 'coverage 94%', 'downloads 91M/month', 'chat on gitter', and 'code helpers 147'. The main description states 'Promise based HTTP client for the browser and node.js' and includes a link to 'New axios docs website: click here'. A 'Table of Contents' lists links to 'Features', 'Browser Support', 'Installing', 'Example', 'Axios API', 'Request method aliases', 'Concurrency (Deprecated)', and 'Creating an instance'. On the right, the 'Install' section shows the command 'npm i axios'. The 'Repository' section points to 'github.com/axios/axios', and the 'Homepage' is 'axios-http.com'. A 'Weekly Downloads' chart shows a value of 20,149,115. Below this, a table lists 'Version' (0.24.0) and 'License' (MIT), and another table lists 'Unpacked Size' (389 kB) and 'Total Files' (48). At the bottom, there are links for 'Issues' and 'Pull Requests', and a dark blue footer bar lists various countries: SAUDI ARABIA, SINGAPORE, SPAIN, SWEDEN, UNITED ARAB EMIRATES, UNITED KINGDOM, and UNITED STATES OF AMERICA.

axios TS

0.24.0 • Public • Published a month ago

Readme Explore BETA 1 Dependency 68,782 Dependents 52 Versions

axios

npm v0.24.0 cdnjs v0.24.0 ci passing Gitpod Ready-to-Code coverage 94% downloads 91M/month chat on gitter code helpers 147

Promise based HTTP client for the browser and node.js

New axios docs website: [click here](#)

Table of Contents

- [Features](#)
- [Browser Support](#)
- [Installing](#)
- [Example](#)
- [Axios API](#)
- [Request method aliases](#)
- [Concurrency \(Deprecated\)](#)
- [Creating an instance](#)

Install

```
> npm i axios
```

Repository

github.com/axios/axios

Homepage

axios-http.com

Weekly Downloads

20,149,115

Version	License
0.24.0	MIT

Unpacked Size	Total Files
389 kB	48

Issues Pull Requests

SAUDI ARABIA SINGAPORE SPAIN SWEDEN UNITED ARAB EMIRATES UNITED KINGDOM UNITED STATES OF AMERICA

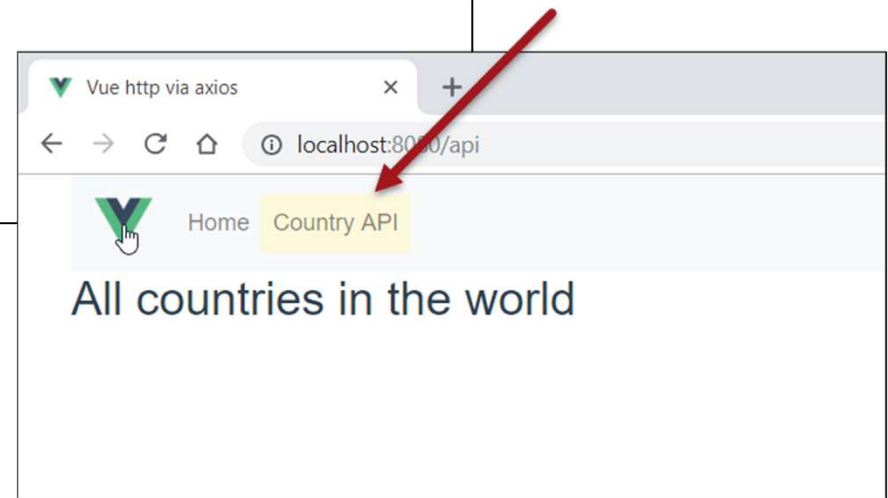
Installing axios

- Add axios to your project
 - `npm install axios --save`
- Import axios in the component where you want to use it.
- It is actually considered *bad practice* to use axios or Http-calls directly from a component!
 - Recommended – use a store mechanism, like Vuex.
 - Or work from a separate Service/API/Data Access layer.

New route

- We're creating a new route and a new component to navigate.
 - Create `ApiComponent.vue`
 - Just an empty component for now
 - Add the route like so:

```
// Component for using to the country API RestCountries  
{  
  path: '/api',  
  name: 'api',  
  component: ApiComponent  
}
```



Using axios

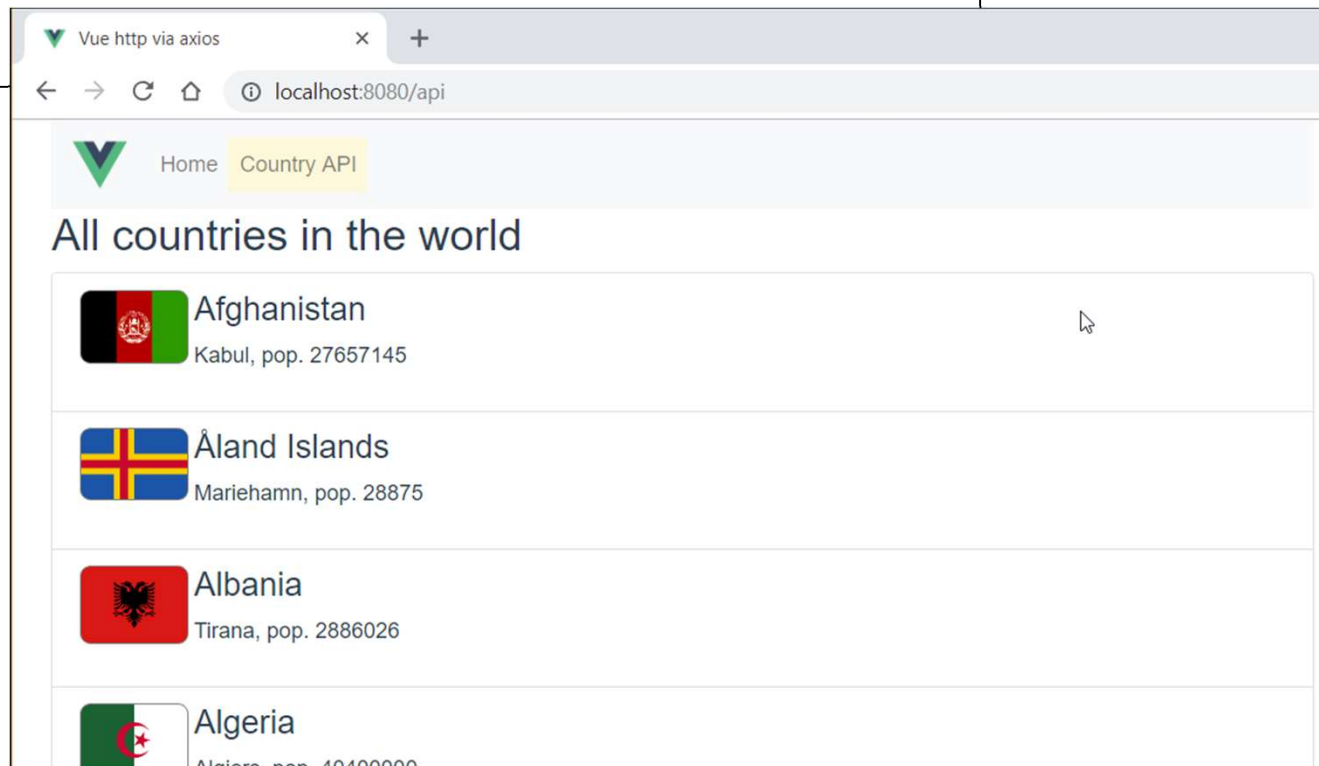
- Call axios inside the component
 - For instance in the `created()` lifecycle hook
 - But can be in an `@click`-method as well...

```
data() {  
  return {  
    // url of the external api  
    url: 'https://restcountries.com/v2/all',  
    countries: [],  
    errors: []  
  }  
},  
created() {  
  // call axios with the url, it returns a promise.  
  axios.get(this.url)  
    .then(response => this.countries = response.data)  
    .catch(err => this.errors.push(err))  
}
```

Update your HTML

```
<ul class="list-group" v-if="countries && countries.length">
  <li class="list-group-item" v-for="country of countries">
    <span class="float-left">
      
    </span>
    <h4>{{ country.name }} </h4>
    <p>{{ country.capital}}, pop. {{ country.population}}</p>
  </li>
</ul>
```

Result



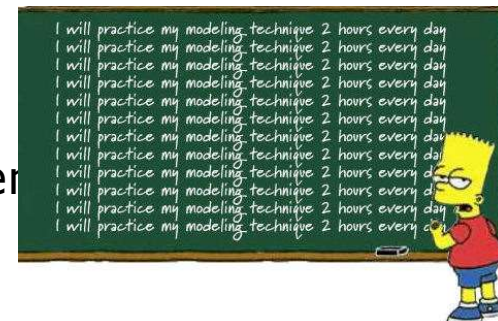
Async/await

- You can use **async/await** with axios
 - Not supported in older browsers (IE 11)
 - Results in the UI are identical

```
async created() {  
    // async / await version (created() becomes async created())  
    try {  
        const response = await axios.get(this.url)  
        this.countries = response.data  
    } catch (err) {  
        this.errors.push(err)  
    }  
}
```

Workshop

- Use your own application, or start from `../330-http-axios`
- Add a new component `<ApiCountryDetail>`
- If a country is selected, go to a country detail page for that country.
- Fetch info for that country over `axios`, show detail component
 - Tip : use the name of `fullname` routes of the API. See <https://restcountries.com/#api-endpoints-v2-name> for details
- Optional – Create an `<ApiSearchCountry>` component, where the user can type in (a part of) the name.
 - Search for that country/countries, show them.
 - When clicked, they should also load the detail component
 - Solution (`../330-workshop`)

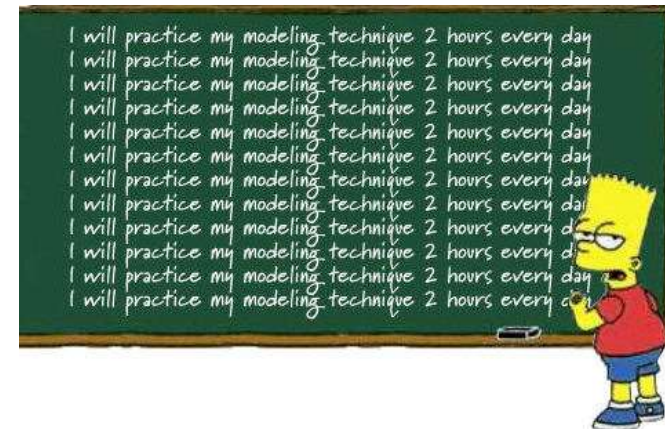


Example - other API's

- <https://pokeapi.co/> - Pokemon API
- <http://openweathermap.org/API> (weather forecast)
- <https://jsonplaceholder.typicode.com/> random users, posts, photos
- <http://ergast.com/mrd/> - Ergast Motor (F1) API
- <http://www.omdbapi.com/> - Open Movie Database
- <https://swapi.dev/> - Star Wars API
- ...many, many more (<https://github.com/public-apis/public-apis>)

Workshop

- Create a small application using one of the API's in the previous slide, for example
 - Pokemon API
 - OMDb API
 - OpenWeatherMap API
 - ...



Checkpoint

- You can perform **http-calls** from your Vue application using axios
- You are able to handle the results from an **external API** and show them in the UI.
- You can do **promise-based** calls and **async/await**-based calls