



Vue Fundamentals – 06

Using Http directly



Peter Kassenaar –
info@kassenaar.com

Peter Kassenaar

INCLUSIEF
GRATIS
WEBVERSIE
VAN HET
BOEK

Vue.js

Web Development Library

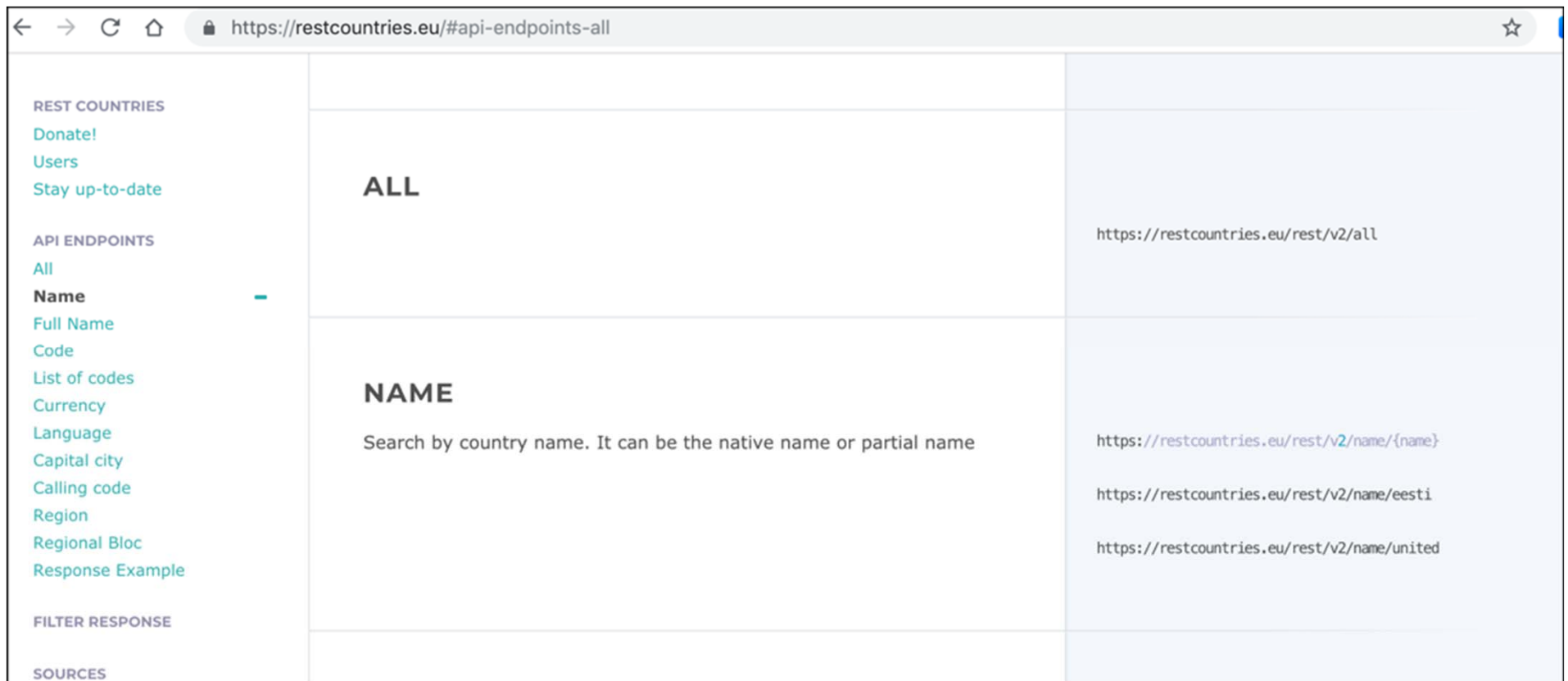


VANDUUREN MEDIA

P. 216 e.v.

Consuming data from an API

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



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>

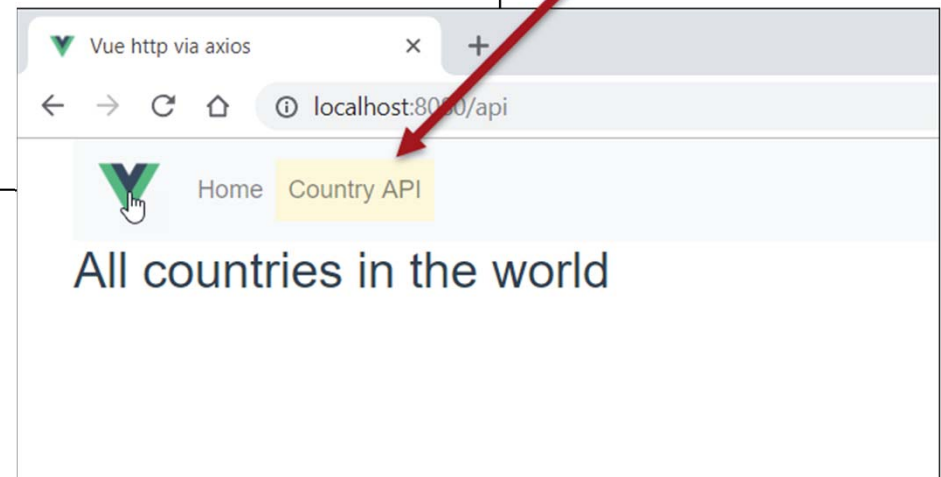
Installing axios

- 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.
 - We're going to cover that next.

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

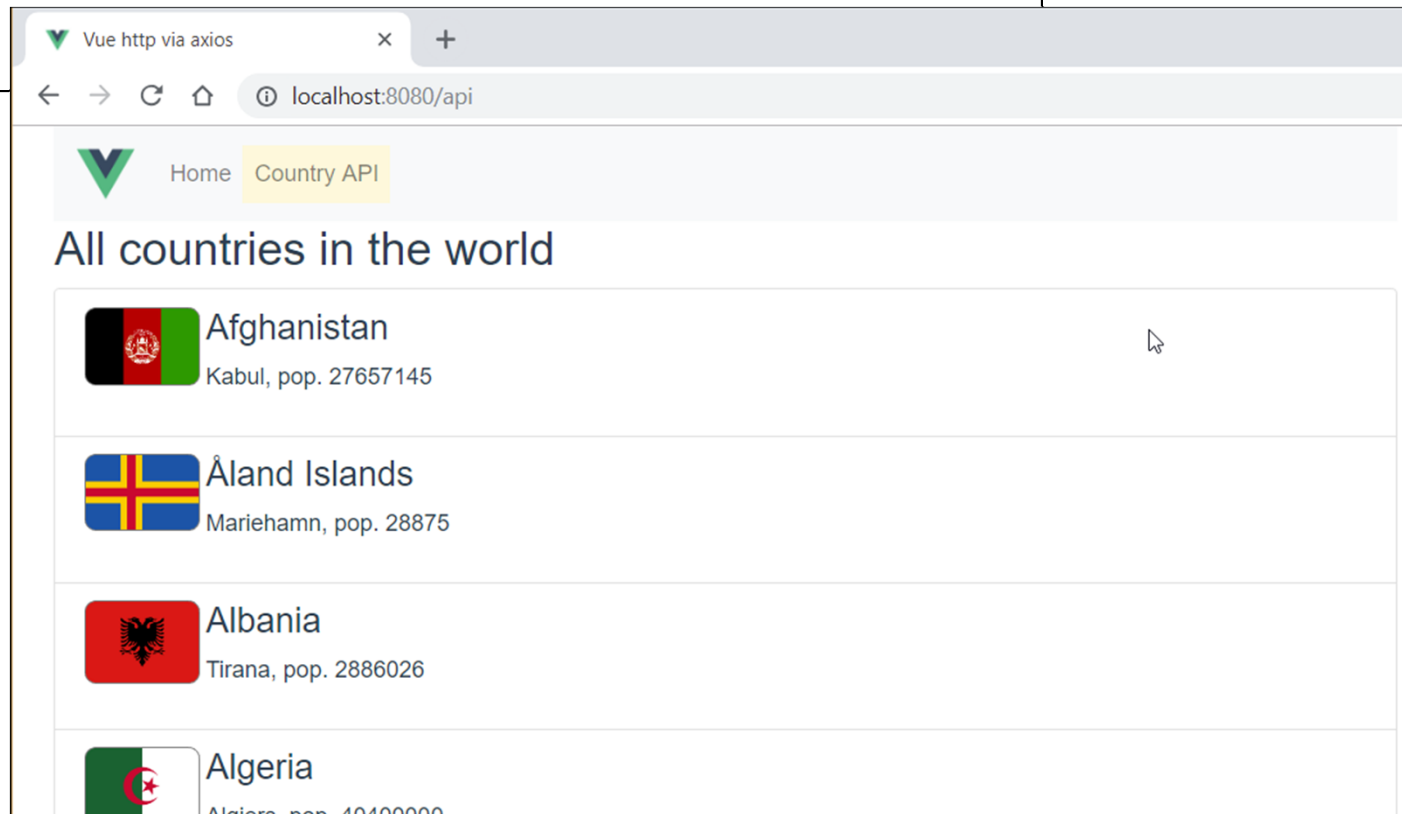
- Call axios inside the component
 - For instance in the `created()` lifecycle hook
 - But can be in an `@click`-method as well...

```
data() {  
  return {  
    // url of the external api  
    url: 'https://restcountries.eu/rest/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



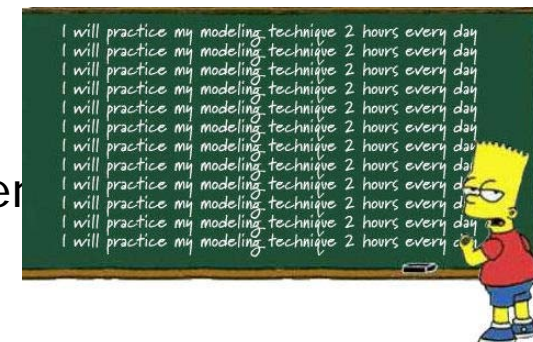
Async/await

- 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.eu/#api-endpoints-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`)



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