

Vue.js

Reden

Tijdens de career day heb ik met verschillende bedrijven gesproken en zij gaven aan dat het belangrijk was om een framework te kennen, waarbij React werd aanbevolen. Dit heb ik met de docent besproken, maar ik kreeg te horen dat het een steile leercurve heeft. Daarom adviseerde hij mij om eerst Vue.js te leren.

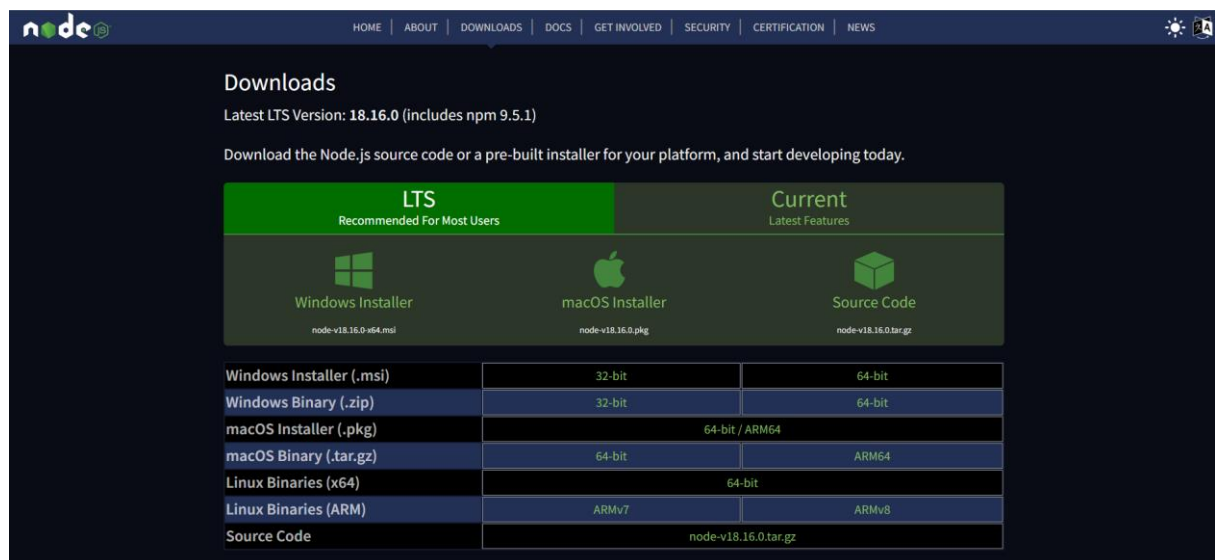
Wat is Vue.js?

Vue.js is een open-source JavaScript-framework dat wordt gebruikt voor het bouwen van gebruikersinterfaces en single-page applicaties. Het is ontworpen om de ontwikkeling van webapplicaties te vergemakkelijken door het bieden van een flexibele en schaalbare structuur.

Installatie

Ik heb Vue.js geïnstalleerd om gebruik te kunnen maken van de mogelijkheden en voordelen die het framework biedt bij het ontwikkelen van webapplicaties. Dit heb ik gedaan door een YouTube video te bekijken.

Bron: <https://www.youtube.com/watch?v=FGNsJhJYaqo>



Als eerst heb ik Node.js geïnstalleerd via <https://nodejs.org/en/download>. Node.js is een JavaScript runtime-omgeving die het mogelijk maakt om JavaScript-code buiten een webbrowser uit te voeren.

```
C:\Users\Talitha>node -v
v18.16.0
```

Door het commando 'node -v' uit te voeren in de opdrachtprompt (CMD), kan ik controleren of Node.js correct is geïnstalleerd en welke versie ervan is geïnstalleerd.

```
C:\Users\Talitha>npm -v
9.5.1
```

npm staat voor Node Package Manager. Het is een krachtige en uitgebreide pakketbeheertool die wordt geleverd bij de installatie van Node.js. npm wordt gebruikt om externe packages, modules en bibliotheken te beheren en te installeren in je Node.js-projecten. Het biedt toegang tot een enorme verzameling open-source packages die zijn gemaakt en bijgedragen door de Node.js-community.

```
npm install -g @vue/cli
# OR
yarn global add @vue/cli
```

Vue.js heb ik geïnstalleerd via <https://cli.vuejs.org/guide/installation.html>.

```
C:\Users\Talitha>vue
Usage: vue <command> [options]

Options:
  -V, --version          output the version number
  -h, --help             display help for command

Commands:
  create [options] <app-name>      create a new project powered by vue-cli-service
  add [options] <plugin> [pluginOptions]  install a plugin and invoke its generator in an already created project
  invoke [options] <plugin> [pluginOptions]  invoke the generator of a plugin in an already created project
  inspect [options] [paths...]  inspect the webpack config in a project with vue-cli-service
  serve                          alias of "npm run serve" in the current project
  build                          alias of "npm run build" in the current project
  ui [options]                  start and open the vue-cli ui
  init [options] <template> <app-name>  generate a project from a remote template (legacy API, requires
                                       @vue/cli-init)
  config [options] [value]        inspect and modify the config
  outdated [options]             (experimental) check for outdated vue cli service / plugins
  upgrade [options] [plugin-name] (experimental) upgrade vue cli service / plugins
  migrate [options] [plugin-name] (experimental) run migrator for an already-installed cli plugin
  info                           print debugging information about your environment
  help [command]                display help for command

Run vue <command> --help for detailed usage of given command.
```

Door het commando 'vue' uit te voeren in de opdrachtprompt (CMD), kan ik controleren of Vue.js correct is geïnstalleerd.

```
C:\Users\Talitha\Desktop\ICT\VueProjects>vue create youtube-oefeningen|
```

Het uitvoeren van het commando 'vue create' is nodig om de officiële Vue-projectstructuurgereedschap te installeren en uit te voeren. Dit gereedschap helpt bij het opzetten van de basisstructuur van een Vue.js-project. Door create-vue te gebruiken, worden alle vereiste bestanden en mappen aangemaakt, zoals de projectconfiguratie, de mapstructuur en basisbestanden zoals de hoofdcomponent en router.

```
App running at:
- Local:   http://localhost:8080/
- Network: http://192.168.0.51:8080/

Note that the development build is not optimized.
To create a production build, run npm run build.
```

Om dit te krijgen, voerde ik het commando 'cd youtube-oefeningen' uit in de opdrachtprompt (CMD). Daarna voerde ik het commando 'npm run serve' uit.



Welcome to Your Vue.js App

For a guide and recipes on how to configure / customize this project,
check out the [vue-cli documentation](#).

Installed CLI Plugins

[babel](#) [eslint](#)

Essential Links

[Core Docs](#) [Forum](#) [Community Chat](#) [Twitter](#) [News](#)

Ecosystem

[vue-router](#) [vuex](#) [vue-devtools](#) [vue-loader](#) [awesome-vue](#)

Wanneer een Vue project gecreëerd wordt, is dit de default webpagina. Deze pagina is aan te passen/configureren.

Link: <http://localhost:8080/>

YouTube oefening

Om te begrijpen hoe Vue.js werkt heb ik een Youtube video bekeken.

Bron: <https://www.youtube.com/watch?v=F7PLPJqVotk&list=WL&index=5>

Vue.js-bibliotheek

```
<script src="https://unpkg.com/vue@3.0.2"></script>
```

De HTML-code `<script src="https://unpkg.com/vue@3.0.2"></script>` verwijst naar een externe bron (URL) waar de Vue.js-bibliotheek wordt gehaald. In dit geval wordt versie 3.0.2 van Vue.js opgehaald van het unpkg.com-cdn. Door Vue.js op deze manier in de HTML-pagina op te nemen, wordt er gebruik van een gehoste versie van de Vue.js-bibliotheek die wordt aangeboden door unpkg.com. Dit betekent dat wanneer de webpagina wordt geladen, de Vue.js-bibliotheek wordt gedownload en uitgevoerd vanaf deze externe bron.

Link: <https://vuejs.org/guide/quick-start.html#using-vue-from-cdn>

Puntkomma's

Vue.js maakt geen gebruik van puntkomma's als afsluiters voor instructies. In Vue.js worden instructies gescheiden door een nieuwe regel of door een speciale syntax genaamd 'Vue-directives' zoals 'v-if', 'v-for', enzovoort. Puntkomma's zijn niet nodig aan het einde van deze instructies.

Console

```
You are running a development build of   vue@3.0.2:9207
Vue.
Make sure to use the production build (*.prod.js) when
deploying for production.
```

Als ik op Inspecteren klik en naar de Console ga, kan ik zien welke versie van Vue.js wordt gebruikt in de huidige build. Zo kan ik checken of de koppeling met de Vue.js-bibliotheek goed werkt.

Vue.createApp

```
const app = Vue.createApp({
```

Vue.createApp is een functie in het Vue.js-framework waarmee je een nieuwe Vue-toepassing kunt maken. Het is een van de belangrijkste functies in Vue.js waarmee je de basisconfiguratie en het gedrag van je app kunt definiëren. Met Vue.createApp kun je een nieuw Vue-applicatie-object maken. Dit object vertegenwoordigt je gehele Vue-toepassing en biedt een omgeving waarin je componenten kunt definiëren, gegevens kunt beheren en de interactie met de gebruiker kunt regelen.

app.mount

```
app.mount('#app') ▶ <div id="app" data-v-app> ... </div>
```

app.mount('#app') betekent dat je de Vue-applicatie, die je hebt gemaakt met behulp van Vue.createApp, aan een specifiek element in de webpagina koppelt. Dit element wordt geïdentificeerd door de selector '#app'.

Variabelen gebruiken

```
<div id="app">
  <p>{{ title }} - {{ author }} - {{ age }}</p>
</div>
```

```
const app = Vue.createApp({
  data() {
    return {
      title: 'The Final Empire',
      author: 'Brandon Sanderson',
      age: 45
    }
  }
})
app.mount('#app')
```

The Final Empire - Brandon Sanderson - 45

Dit betekent dat de waarden van de title, author en age in het gegevensobject worden weergegeven in de <p>-tag binnen de <div>-tag met de id "app".

```
<div id="app">
  <p>{{ title }} - {{ author }} - {{ age }}</p>
  <button v-on:click="age++">Increase age</button>
  <button v-on:click="age--">Decrease age</button>
</div>
```

The Final Empire - Brandon Sanderson - 32

Increase age Decrease age

De eerste knop heeft het attribuut v-on:click dat een event handler toewijst wanneer erop wordt geklikt. In dit geval is de event handler age++, wat betekent dat de waarde van de age-variabele met één wordt verhoogd wanneer erop wordt geklikt. De tweede knop heeft ook het attribuut v-on:click dat een event handler toewijst. In dit geval is de event handler age--, wat betekent dat de waarde van de age-variabele met één wordt verlaagd wanneer erop wordt geklikt.

```
<div id="app">
  <p>{{ title }} - {{ author }} - {{ age }}</p>
  <button v-on:click="age++">Increase age</button>
  <button v-on:click="age--">Decrease age</button>
  <div @click="changeTitle">Change book title</div>
</div>
```

```
const app = Vue.createApp({
  data() {
    return {
      title: 'The Final Empire',
      author: 'Brandon Sanderson',
      age: 45
    }
  },
  methods: {
    changeTitle() {
      this.title = 'Words of Radiance'
    }
  }
})
app.mount('#app')
```

@click betekent hetzelfde als v-on:click.

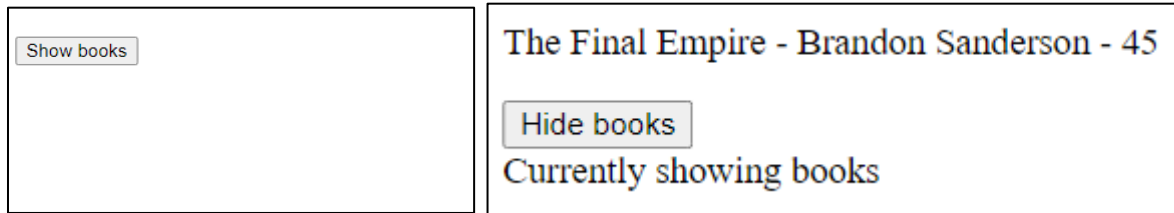
Words of Radiance - Brandon Sanderson - 45

Increase age Decrease age

Change book title

Deze code bevat een HTML-element <div> met een @click-attribuut dat een event handler toewijst aan de klikgebeurtenis op dat element. Wanneer erop wordt geklikt, wordt de event handler changeTitle geactiveerd. In dit geval wordt de methode changeTitle uitgevoerd. De changeTitle-methode maakt deel uit van een Vue-component en wordt gedefinieerd binnen een methods-object. Deze methode verandert de waarde van de title-eigenschap in 'Words of Radiance'.

Show and hide



```
<div id="app">
  <div v-if="showBooks">
    <p>{{ title }} - {{ author }} - {{ age }}</p>
  </div>

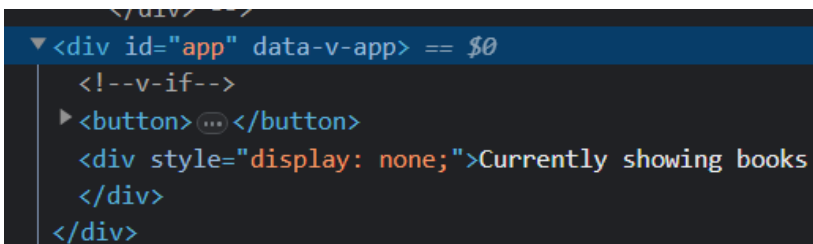
  <button @click="toggleShowBooks">
    <span v-if="showBooks">Hide books</span>
    <span v-else>Show books</span>
  </button>

  <div v-show="showBooks">Currently showing books</div>
</div>
```

Binnen de div-container is er een ander div-element met 'v-if'. De 'v-if' controleert de waarde van de showBooks-variabele in de data van de Vue.js-applicatie. Als showBooks true is, wordt het div-element weergegeven en bevat het een p-element met gegevens zoals title, author en age.

Daarna is er een button-element met de @click-directive. Dit luistert naar het click-evenement en roept de methode toggleShowBooks aan wanneer erop wordt geklikt. Binnen het button-element zijn twee span-elementen met de v-if- en v-else-directives. Deze worden gebruikt om dynamisch tekst weer te geven op basis van de waarde van showBooks. Als showBooks true is, wordt de eerste span weergegeven met de tekst "Hide books". Als showBooks false is, wordt de tweede span weergegeven met de tekst "Show books".

Tot slot is er een ander div-element met 'v-show'. Deze directive controleert ook de waarde van showBooks. Als showBooks true is, wordt het div-element weergegeven met de tekst "Currently showing books". Als showBooks false is, wordt het div-element verborgen maar neemt het nog steeds ruimte in op de pagina.



Wanneer de 'v-if' directive wordt gebruikt en de waarde ervan 'false' is, wordt de 'div showBooks' uit de DOM verwijderd. Het toevoegen en verwijderen van elementen uit de DOM vereist meer tijd en geheugen dan simpelweg gebruik te maken van CSS.

Aan de andere kant, wanneer de 'v-show' directive wordt gebruikt, wordt de 'div' verborgen door middel van CSS met behulp van de 'display: none' of weergegeven door middel van de 'display: block' - eigenschap. Hierdoor blijft de 'div' bij gebruik van 'v-show' nog steeds ruimte innemen op de pagina.

```
const app = Vue.createApp({
  data(){
    return {
      showBooks: true,
      title: 'The Final Empire',
      author: 'Brandon Sanderson',
      age: 45
    }
  },
  methods: {
    toggleShowBooks() {
      this.showBooks =! this.showBooks
    }
  }
})

app.mount('#app')
```

Deze code maakt een Vue.js-applicatie met een gegevensobject dat onder andere een boolean genaamd 'showBooks' bevat. Er is ook een methode genaamd 'toggleShowBooks' die de waarde van 'showBooks' omkeert wanneer deze wordt aangeroepen.