

# Full Stack Web Development

## WS2019 - BSA5 Ausgewählte Kapitel

Alija Sasic

[sasic@technikum-wien.at](mailto:sasic@technikum-wien.at)

Smart Homes and Assistive Systems

October 2, 2019

# Outline

## 1 Vue

1.1 Intro

1.2 Tools

1.3 Vue Instance

1.3.1 data

1.3.2 methods

1.3.3 computed

1.3.4 watcher

1.4 Directives

1.4.1 v-if

1.4.2 v-model

1.4.3 v-bind

1.4.4 v-on

1.5 Workflow

# Vue - JavaScript Frameworks

- AngularJS
- Angular
- React
- Vue.js
- Ember.js
- ...



Alija Sabic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Evan You [1, 3]

- Born and raised in Wuxi, PRC (无锡中国).
- Recruited by Google for his work, inspired by [Chrome experiments](#), and joined the [Five program](#) during his study in the U.S.
- Started the work on [Vue.js](#) as a personal project during his work at Google Creative Labs in 2013.
- Works full-time on Vue since 2016.

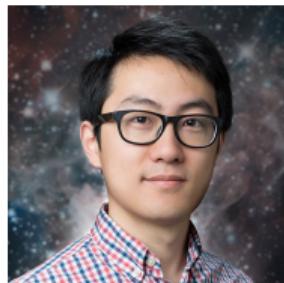


Figure 1: Evan You  
(尤雨溪)



Alija Sasic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue.js

- Client-side JavaScript framework
- Progressive framework
- Performant
- Lightweight (~33.30kB - min+gzip)
- Extensible
  - Vue Router
  - Vuex
  - Vue Server Renderer



Figure 2: Vue.js Logo  
[2]



Alija Sabic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue.js - Core Features



Figure 3: Vue.js Core Features



Alija Sabic

.....  
Web Development.....

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Hello Vue



Vue applications can be created in different ways. To get started, we will employ a simpler setup and add the Vue script in the <head> section, using a <script> element.

```
<script src="https://vuejs.org/js/vue.js"></script>
```

Listing 1: Adding Vue to your page.

- Alija Sabic  
Web Development
- Vue
- Intro
  - Tools
  - Vue Instance
  - data
  - methods
  - computed
  - watcher
  - Directives
  - v-if
  - v-model
  - v-bind
  - v-on
  - Workflow

# Vue - Hello Vue

With the addition of this script, we can now access the class `Vue` in our scripts.

To create an application, we have to instantiate a object of this class. We have to pass a `JSON` object to the class constructor to configure the application.

```
<script>
  new Vue({
    el: "#app",
    data: {
      text: "Hello Vue!"
    }
  });
</script>
```

Listing 2: Creating a Vue application.

Vue uses the passed `JSON` object to setup the application.



Alija Sasic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Hello Vue

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <meta charset="utf-8" />
5          <title>Vue Setup</title>
6          <script src="https://vuejs.org/js/vue.js"></script>
7      </head>
8      <body>
9          <div id="app">{{ text }}</div>
10     </body>
11     <script>
12         new Vue({
13             el: "#app",
14             data: {
15                 text: "Hello Vue!"
16             }
17         });
18     </script>
19 </html>
```

Listing 3: Minimal Example



Alija Sasic

..... Web Development .....

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Hello Vue

We use the `el` property to specify what element should be used as the root element of our application. Vue interprets this parameter as a `CSS` selector to find the specified element(s) in the `DOM` of our page.

Vue then uses the specified element(s) as a *template* to create a virtual `DOM`, which is then used internally to create a new (static) `DOM`. Subsequently, Vue replaces the specified element(s) with the newly created `DOM` tree.

We can use other properties in the `JSON` object to customize our application.

In this example, we use the `data` property to specify the data of our application. We can then access the data in the (virtual) `DOM` (*template syntax*).

We will explore the `data` and other properties in more detail, in the following sections.



Alija Sasic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Hello Vue

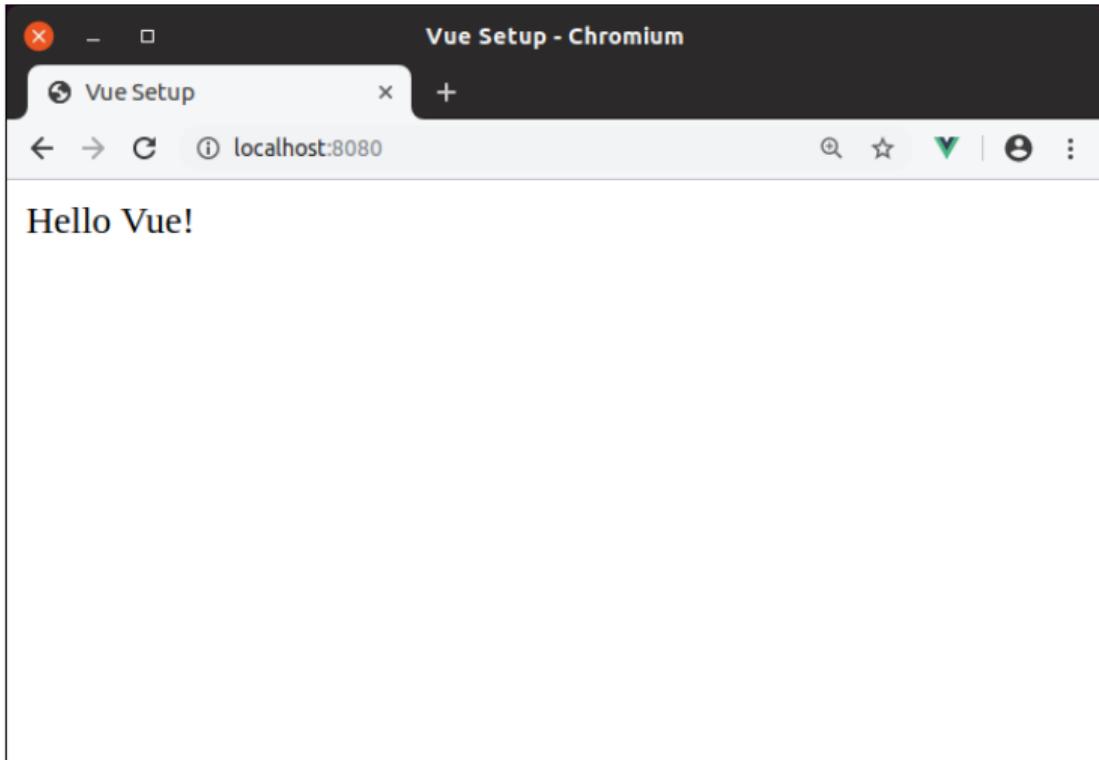


Figure 4: Minimal Example



Alija Sabic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Hello Vue

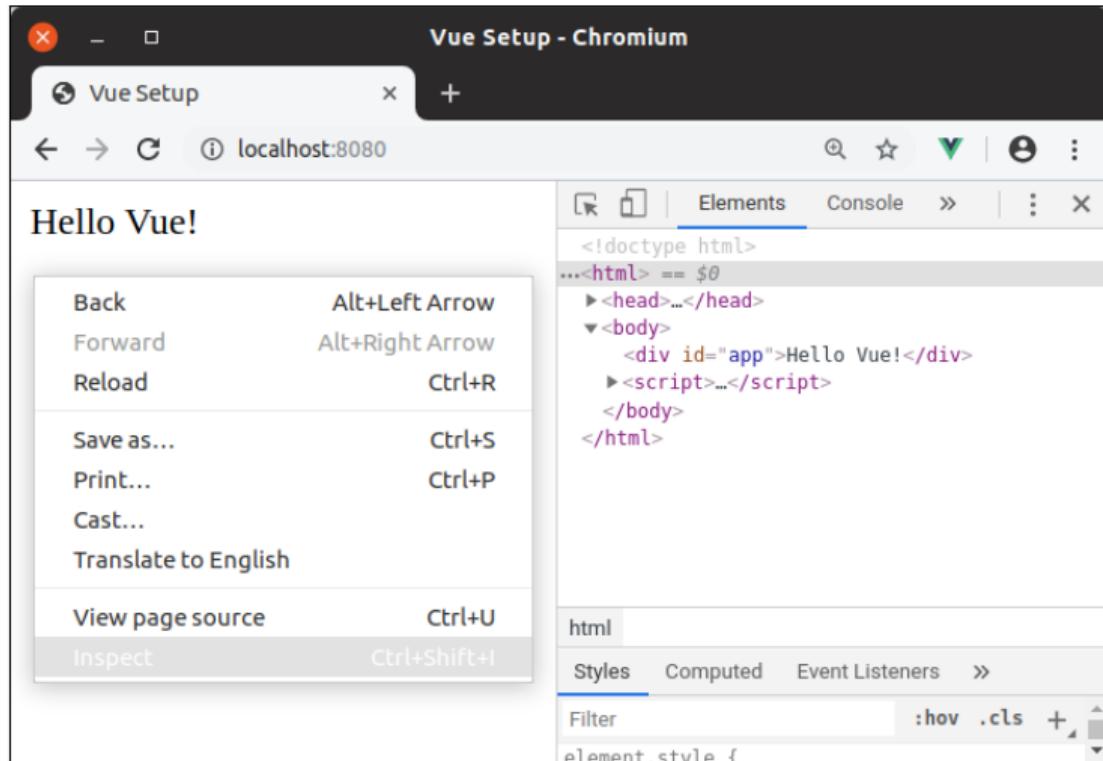


Figure 5: Chromium Developer Tools



Alija Sabic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Browser Devtools

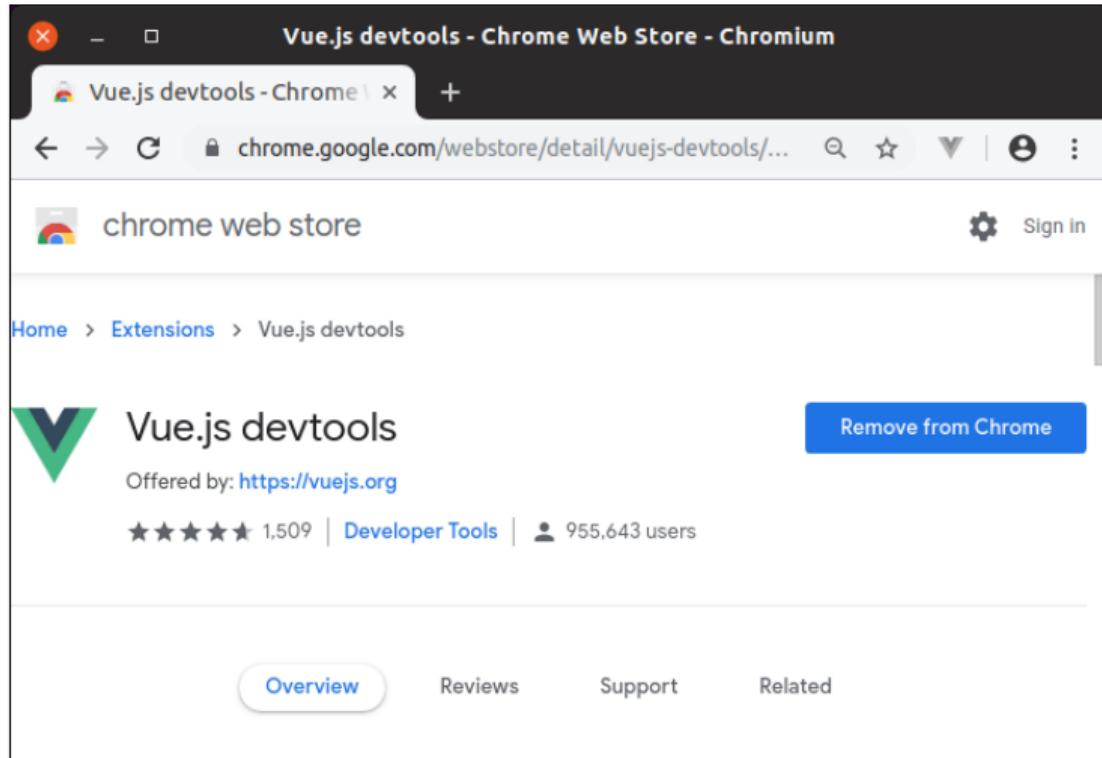


Figure 6: Chrome Vue.js Devtools



Alija Sasic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Browser Devtools

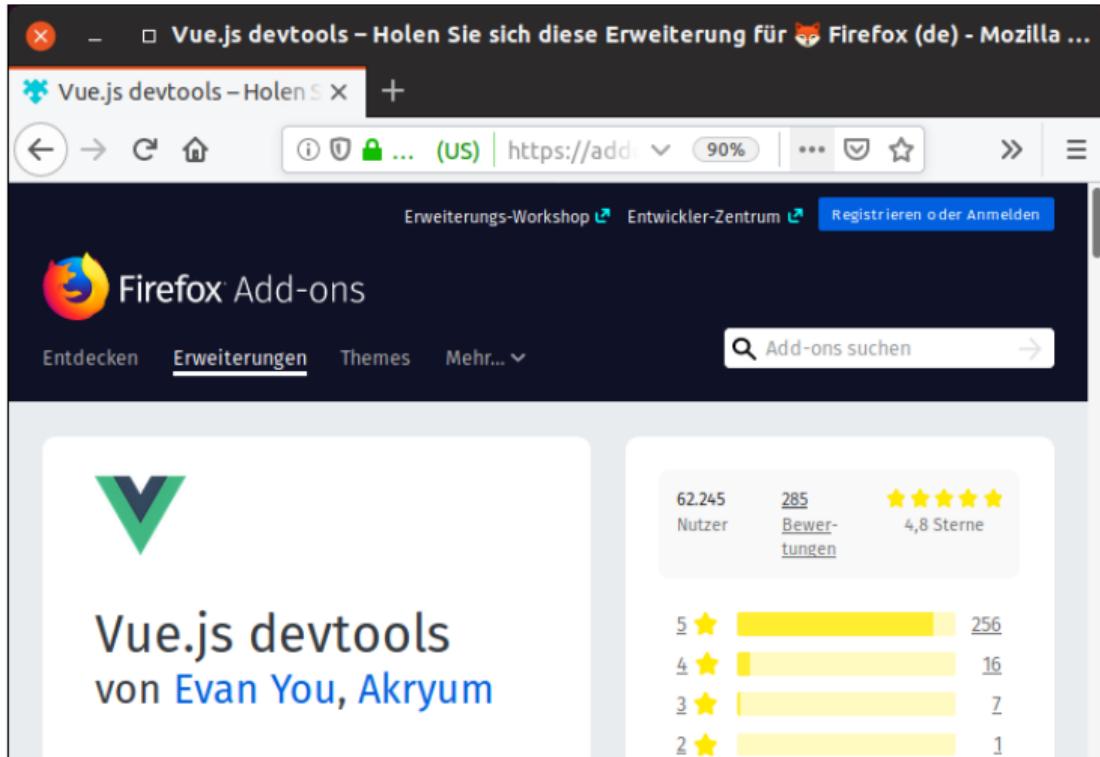


Figure 7: Firefox Vue.js Devtools



Alija Sasic  
Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Browser Devtools

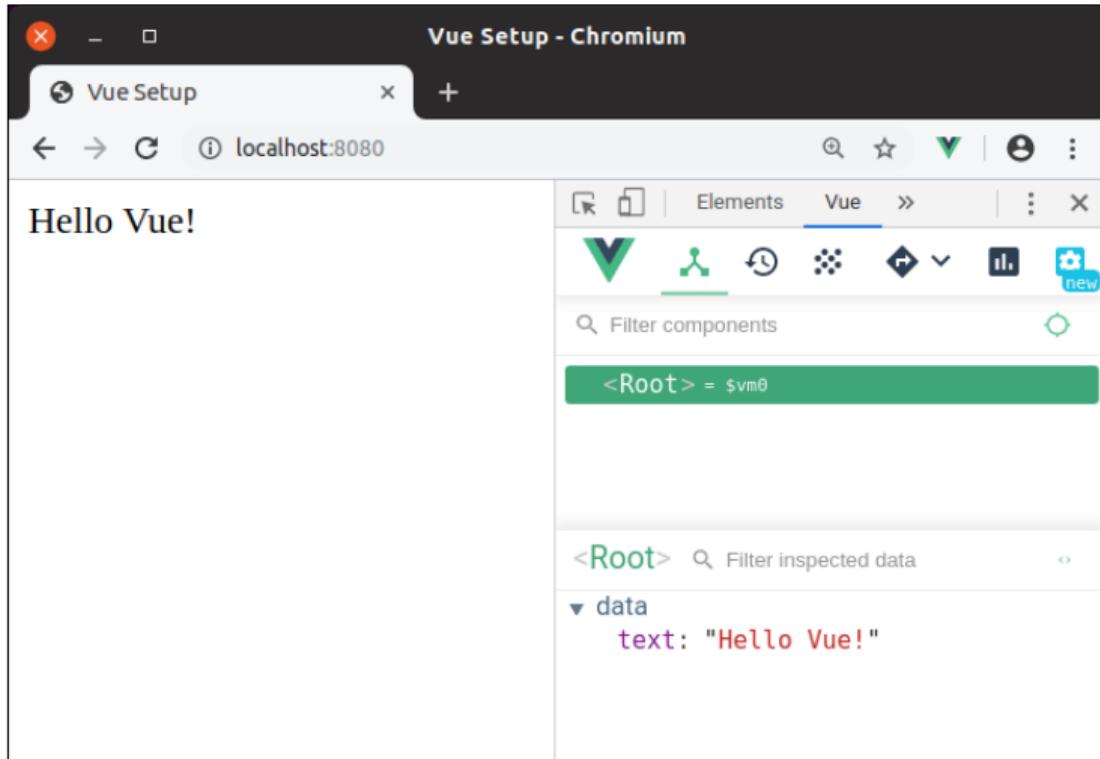


Figure 8: Chrome Vue.js Devtools - Tab



Alija Sasic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - HTTP Server



```
1 $~ node -v
2 v10.16.3
3 $~ npm -v
4 6.9.0
5 $~ npm install -g http-server
6 + http-server@0.11.1
7 added 26 packages from 28 contributors in 6.096s
```

Listing 4: Install [npm](#) module `http-server`

Alija Sabic

[Web Development](#)

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - HTTP Server



```
1 $~ ls
2 Desktop Downloads Pictures Public Videos
3 Documents Music project Templates
4 $~ http-server project/
5 Starting up http-server, serving project/
6 Available on:
7   http://127.0.0.1:8080
8 Hit CTRL-C to stop the server
9 $~/project http-server .
10 Starting up http-server, serving .
11 Available on:
12   http://127.0.0.1:8080
13 Hit CTRL-C to stop the server
```

**Listing 5:** Host directory with http-server

Alija Sasic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Reactivity

Vue enables to declaratively render data to the **DOM** using *template syntax*.

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <meta charset="utf-8" />
5      <title>Vue Setup</title>
6      <script src="https://vuejs.org/js/vue.js"></script>
7    </head>
8    <body>
9      <div id="app">
10        <p>{{ text }}</p>
11      </div>
12    </body>
13    <script>
14      new Vue({
15        el: "#app",
16        data: {
17          text: "Hello Vue!"
18        }
19      });
20    </script>
21  </html>
```

Listing 6: DOM reactivity.



Alija Sabic

.....  
Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Reactivity

Even though, the example is pretty simple and straightforward, Vue does a lot of work in the background. Vue's `data` property and the `DOM` are now linked together, which results in immediate updates of the `DOM` when changing the used properties.

To test the *reactivity* of our application, we can save a reference to our Vue instance and change the property manually using the browser's console.

```
13 <script>  
14   const app = new Vue({  
15     el: "#app",  
16     data: {  
17       text: "Hello Vue!"  
18     }  
19   });  
20 </script>
```

**Listing 7:** Saving a reference to the Vue instance.



Alija Sasic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Reactivity

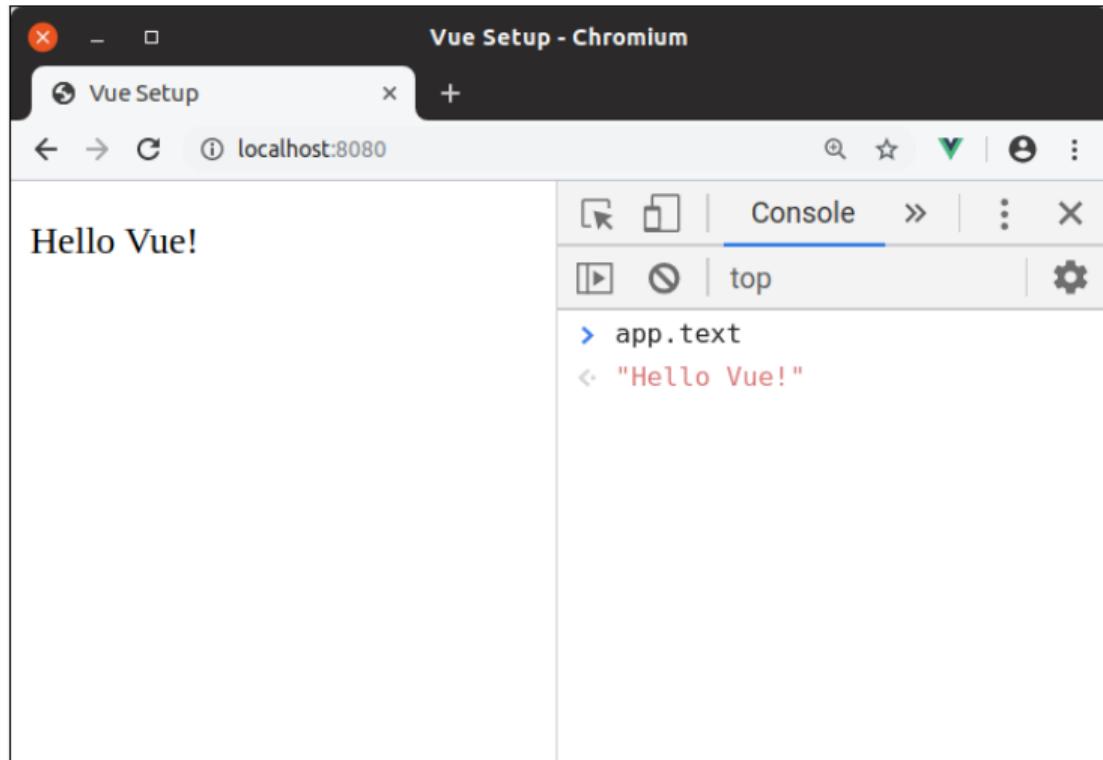


Figure 9: Changing properties of the Vue instance.



Alija Sasic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Reactivity

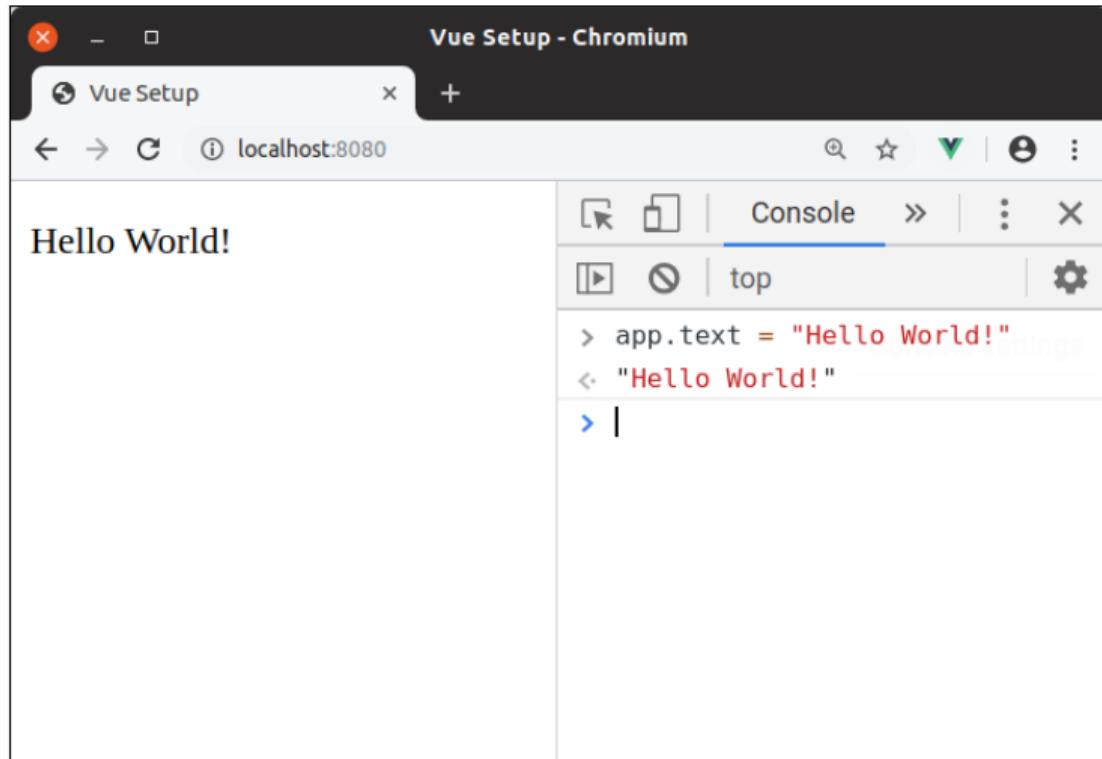


Figure 9: Changing properties of the Vue instance.



Alija Sasic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Property data



The `data` property will be used by Vue to create getters and setters from the properties inside of it.

```
14
15   data: {
16     text: "Hello Vue!"
17   },
```

**Listing 8:** The `data` property of the `JSON` (configuration) object.

Alija Sabic  
Web Development

Vue

Intro

Tools

Vue Instance

**data**

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Property data

We can then access these properties (e.g., `text`)

- Inside the (virtual) `DOM`, using *Interpolation* or *String Interpolation*.
- Inside the Vue instance, using a (virtual) `this` reference.
- Using a reference to the Vue instance, either by
  - Vue's internal properties, that is
    - ▶ `app.$data.text` or
    - ▶ `app._data.text`
  - Generated getters and setters, that is `app.text`



Alija Sabic

Web Development

Vue

Intro

Tools

Vue Instance

**data**

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Property data

```
9 <div id="app">{{ text }}</div>
```

Listing 9: (String) Interpolation inside the (virtual) DOM.

```
12 const app = new Vue({  
13   el: "#app",  
14   data: {  
15     text: "Hello Vue!"  
16   },  
17   methods: {  
18     hello() {  
19       console.log(this.text);  
20     }  
21   }  
22 });
```

Listing 10: Accessing data inside the Vue instance.

```
23 console.log(app.text);  
24 console.log(app.$data.text);
```

Listing 11: Accessing the data with a reference to the Vue instance.



Alija Sabic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Vue - Property data

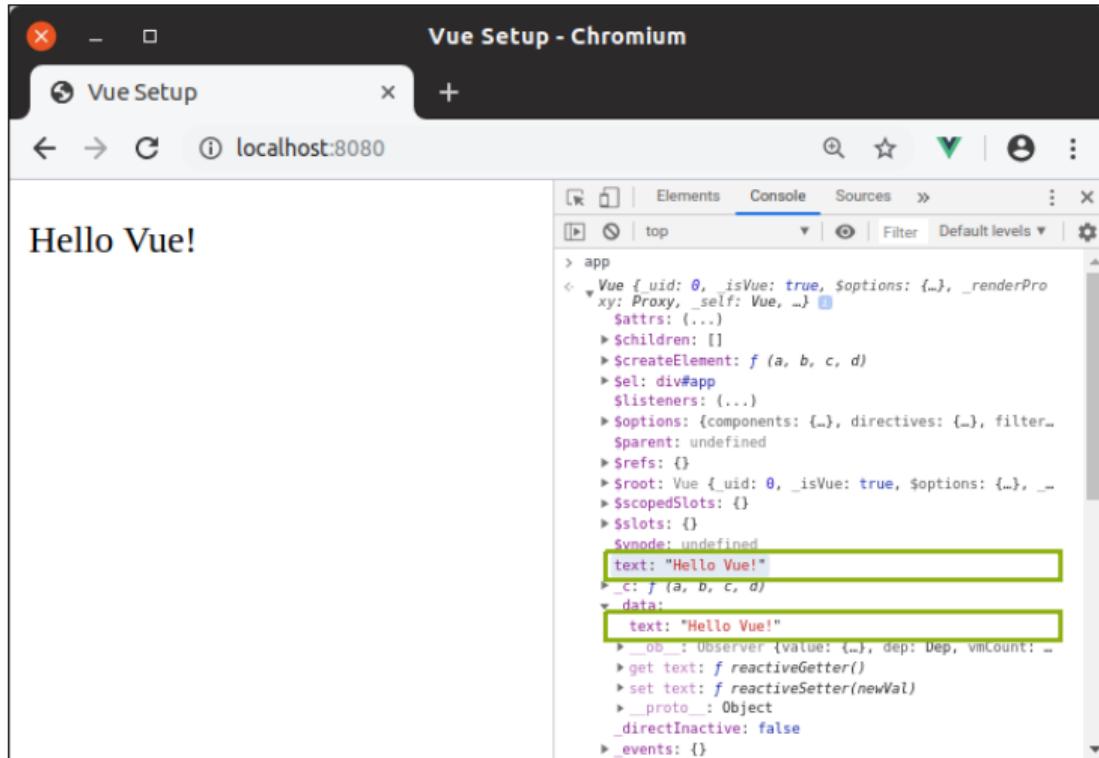


Figure 10: Getters and setters of the Vue instance.



Alija Sasic

Web Development

Vue

Intro

Tools

Vue Instance

data

methods

computed

watcher

Directives

v-if

v-model

v-bind

v-on

Workflow

# Acronyms I



CSS Cascading Stylesheets

DOM Document Object Model

HTTP Hypertext Transfer Protocol

JSON JavaScript Object Notation

npm Node Package Manager

PRC People's Republic of China

U.S. United States

Alija Sabic

Web Development

Acronyms

References

# References I

- [1] V. Cromwell, "Between the Wires: An interview with Vue.js creator Evan You," <https://dev.to/betweenthewires/evan-you>, 11 2016.
- [2] Evan You, "The Progessive JavaScript Framework," <https://vuejs.org/>, 2014.
- [3] ——, "GitHub Customer Stories - Evan You," <https://github.com/customer-stories/yyx990803>, 2019.



Alija Sasic

..... Web Development .....

Acronyms

References