



# Vue 2

Petr Ptáček

# Osnova

- Co to je Vue.js
- Historie
- Porovnání s ostatními frameworky
- Vytvoření aplikace
- Základní konstrukce
- Pohled na ukázkovou aplikaci

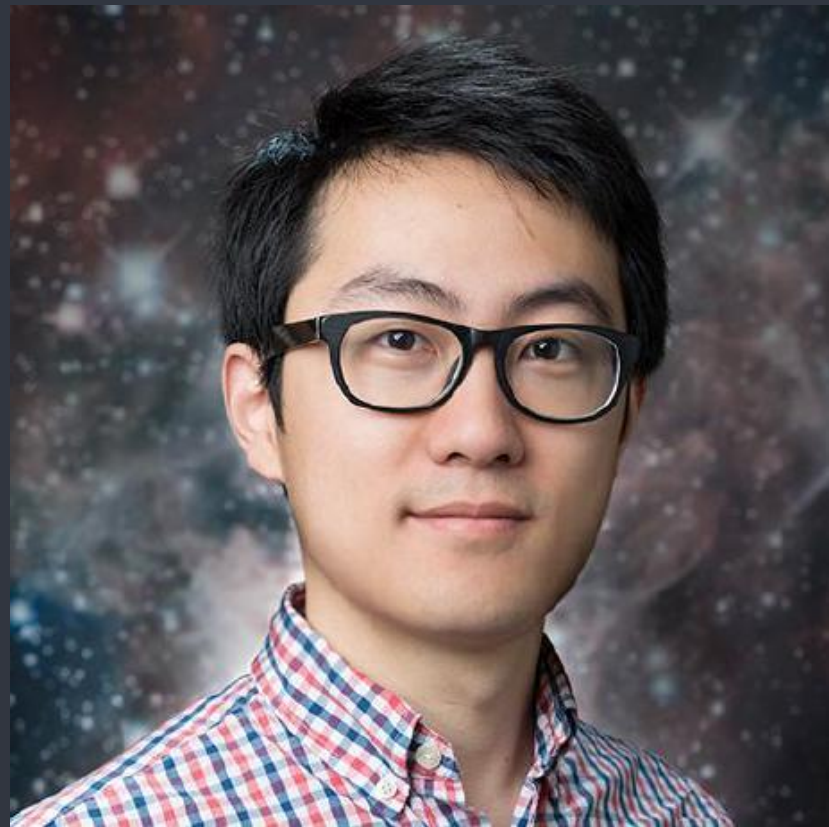
# Co to je Vue.js

- open-source JavaScript progressive framework
- webový framework ke tvoření interaktivních front-end aplikací
- Single-Page Applications v kombinaci s dalšími tooly a podporovanými knihovnami
- jednoduchá integrace s ostatními knihovnami
- [Vue.js The Documentary \(Trailer\)](#)
- [Vue.js The Documentary](#)

# Evan You

<https://github.com/yyx990803>

- project lead of Vue.js
- Singapur
- Creative Technologist (Google)
- od 2016 full-time on Vue.js framework



# Historie

- první zdrojové kódy - červen 2013
- release version - únor 2014
- v1.0.0 - říjen 2015
- v2.6 - únor 2017
- v3.0 - září 2020
- current version of v2 is ^2.7 (final minor version of vue 2.x, Vue2 reach end of life on December 2023)

Version	Release date	Title
0.6	December 8, 2013	VueJS
0.7	December 24, 2013	N/A
0.8	January 27, 2014	N/A
0.9	February 25, 2014	Animatrix
0.10	March 23, 2014	Blade Runner
0.11	November 7, 2014	Cowboy Bebop
0.12	June 12, 2015	Dragon Ball
1.0	October 27, 2015	Evangelion
2.0	September 30, 2016	Ghost in the Shell
2.1	November 22, 2016	Hunter X Hunter
2.2	February 26, 2017	Initial D
2.3	April 27, 2017	JoJo's Bizarre Adventure
2.4	July 13, 2017	Kill la Kill
2.5	October 13, 2017	Level E
2.6	February 4, 2019	Macross
3.0	September 18, 2020	One Piece

# Silné stránky

- Very small Size (20 kB min + gzip)
  - Vue v3 (< 10 kB)
  - React + react-dom 34.4 kB
  - Angular 62 kB
  - Vue.js 23.8 kB
  - jQuery 31 kB
- učící křivka (lehký k pochopení a vývoj aplikací)
- detailní dokumentace <https://v2.vuejs.org/>
- komponentově orientovaný framework

# Ecosystem

- vue-router <https://router.vuejs.org/>
  - Single-page application routing
- vuex <https://vuex.vuejs.org/>
  - state management
- vue-cli <https://cli.vuejs.org/> vite <https://vitejs.dev/>
  - project scaffolding
- vue-loader
  - Single file component (\*.vue file) loader for webpack
- vue-class-component
  - TypeScript decorator for a class-based API
- vue-rx
  - RxJS integration
- vue-devtools
  - Browser DevTools extension

# UI Frameworks

- Vuetify <https://vuetifyjs.com/en/>
- Semantic UI <https://semantic-ui-vue.github.io/#/>
- Bootstrap Vue <https://bootstrap-vue.org/>

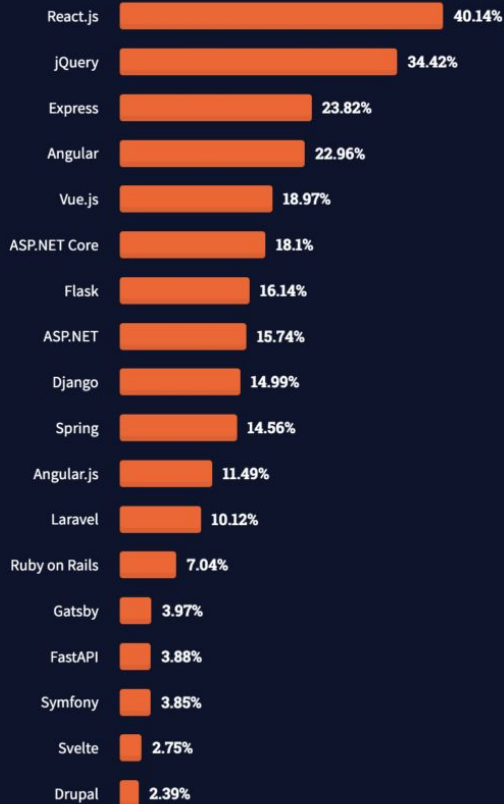


# Kdo to používá

- GitLab
- Baidu
- Netflix
- Trivago
- Facebook
- Google
- Apple
- Nintendo
- Alibaba

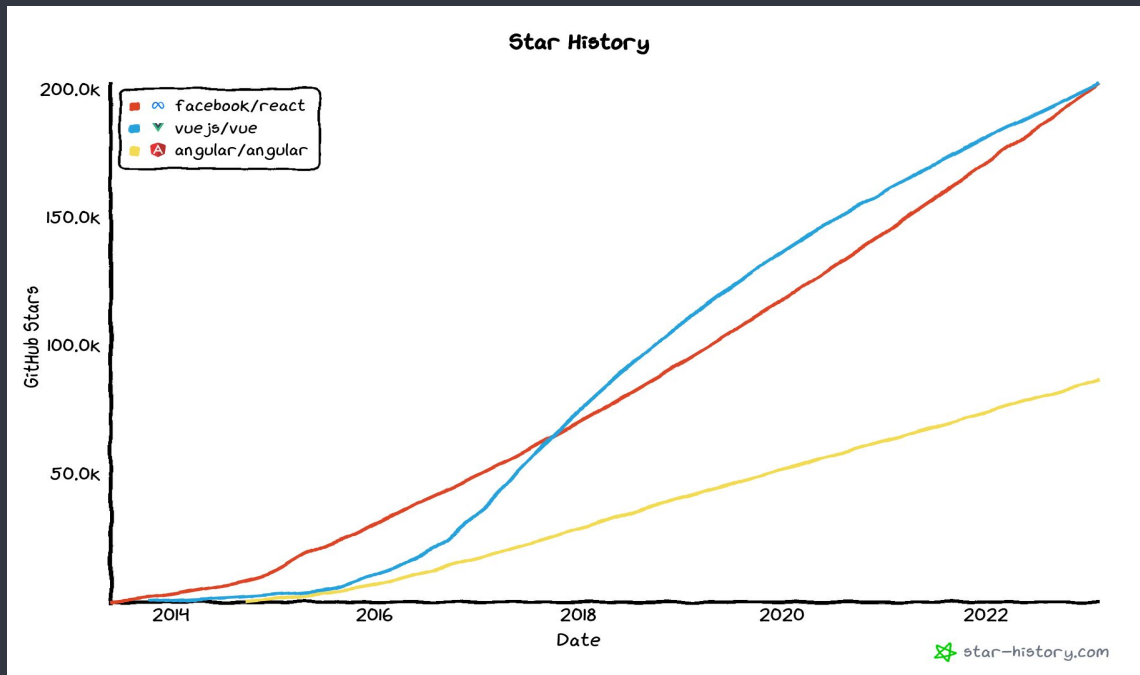
# Framework popularity

<https://survey.stackoverflow.co/2022/#section-most-popular-technologies-web-frameworks-and-technologies>

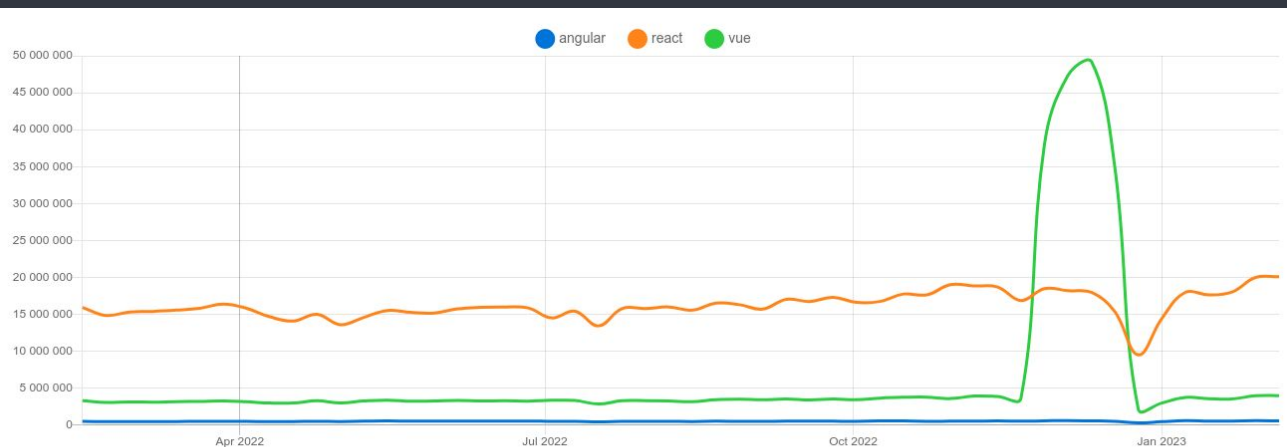


# Star History









<https://star-history.com/#facebook/react&vuejs/vue&angular/angular&Date>



# Downloads

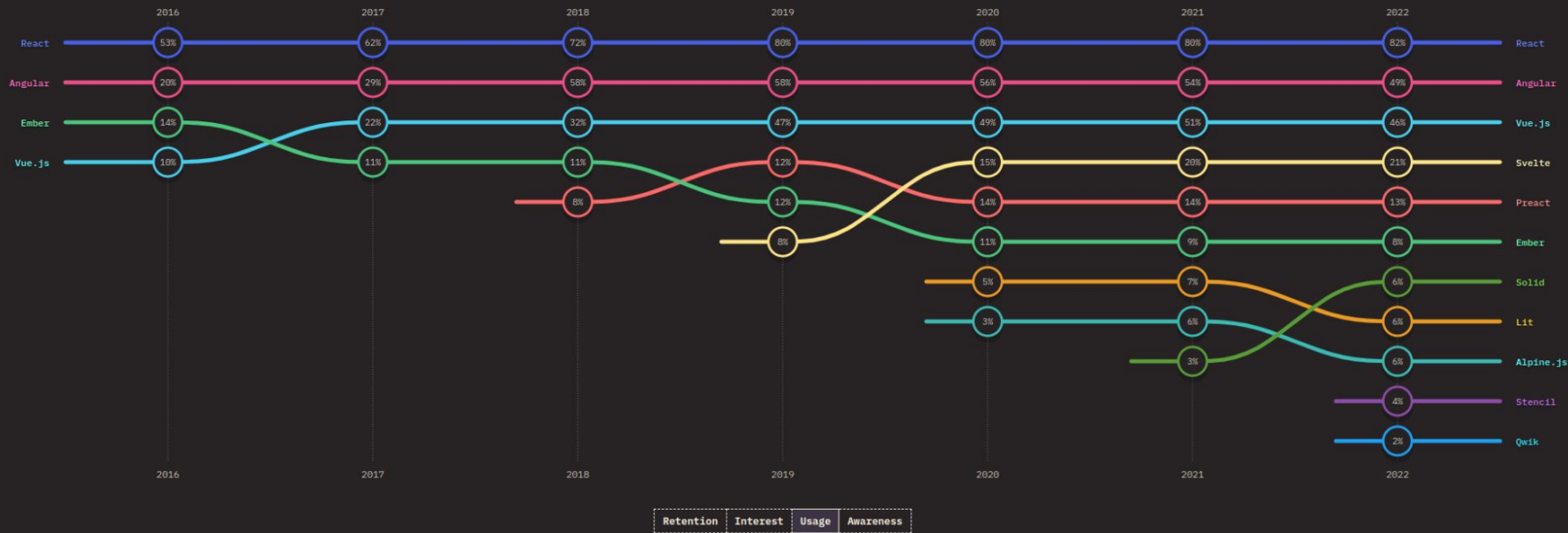


## Stats

			Stars	Issues	Version	Updated <sup>Ⓢ</sup>	Created <sup>Ⓢ</sup>	Size
	angular	  	59 272	467	1.8.3	10 months ago	11 years ago	minzipped size <b>62.3 KB</b>
	react	  	202 153	1 169	18.2.0	8 months ago	11 years ago	minzipped size <b>2.5 KB</b>
	vue	  	35 269	882	3.2.47	13 days ago	9 years ago	minzipped size <b>34.1 KB</b>

<https://npx.trends.com/angular-vs-react-vs-vue>

# 2022 State of JS



# Vytvoření aplikace

1. Node.js (<https://nodejs.dev/en/>)
2. > npm create vue@2
3. instalace devtools
  - a. <https://devtools.vuejs.org/guide/installation.html>

```
~/Test
$ npm create vue@3

Vue.js - The Progressive JavaScript Framework

✓ Project name: ... vue-project
✓ Add TypeScript? ... No / Yes
✓ Add JSX Support? ... No / Yes
✓ Add Vue Router for Single Page Application development? ... No / Yes
✓ Add Pinia for state management? ... No / Yes
✓ Add Vitest for Unit Testing? ... No / Yes
✓ Add an End-to-End Testing Solution? > Cypress
✓ Add ESLint for code quality? ... No / Yes
✓ Add Prettier for code formatting? ... No / Yes

Scaffolding project in /Users/haoqun/Test/vue-project...

Done. Now run:

  cd vue-project
  npm install
  npm run lint
  npm run dev
```

# Náš playground

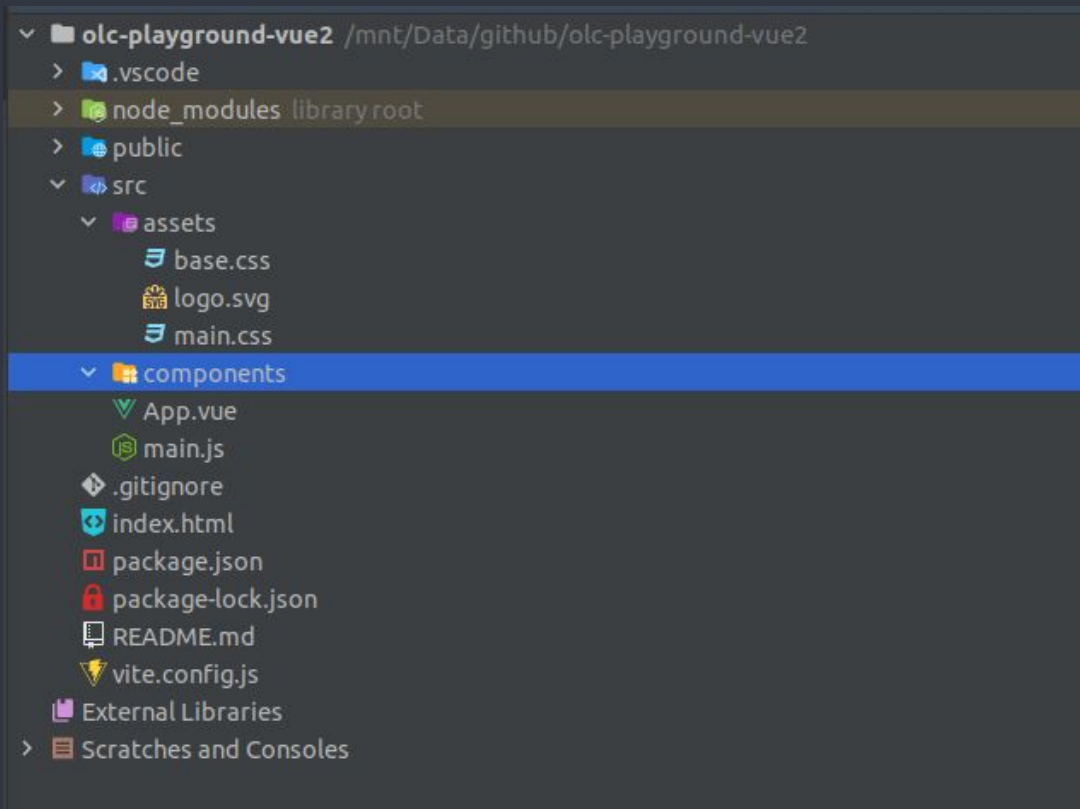
<https://github.com/petr-ptacek/olc-playground-vue2.git>

45

The image displays a grid of 10 counter components, each with a unique ID and a light blue or green background. Each counter shows a current value, a 'Doubled' value, and two buttons: 'Increment' and 'Decrement'. The counters are arranged in two rows of six. The top row (blue) contains CounterID 0 to 5, and the bottom row (green) contains CounterID 6 to 9. CounterID 1, 3, 5, 7, and 9 have an orange 'X' icon in the top right corner.

CounterID	Current Value	Doubled Value	Has Error (X)
CounterID 0	0	0	No
CounterID 1	1	2	Yes
CounterID 2	2	4	No
CounterID 3	3	6	Yes
CounterID 4	4	8	No
CounterID 5	5	10	Yes
CounterID 6	6	12	No
CounterID 7	7	14	Yes
CounterID 8	8	16	No
CounterID 9	9	18	Yes

# Struktura projektu





# Vue instance a \*.vue file

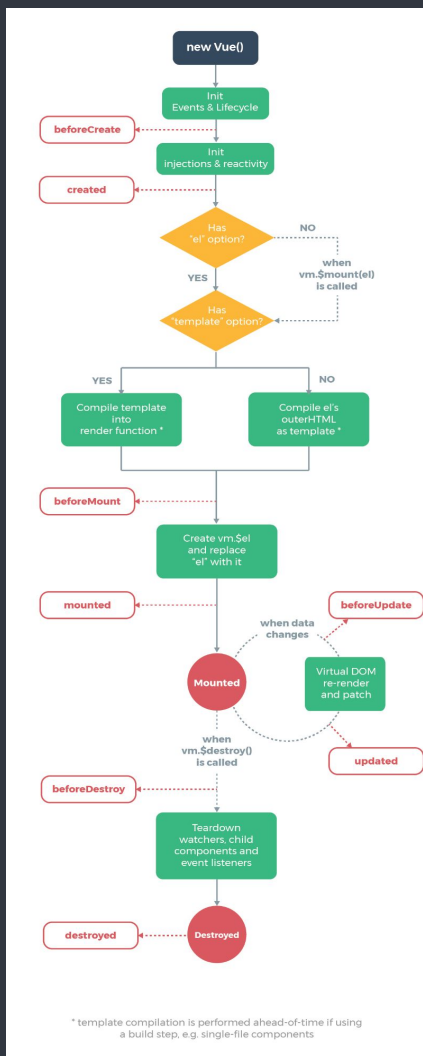
Vue Single-File Components (a.k.a. \*.vue files, abbreviated as SFC) is a special file format that allows us to encapsulate the template, logic, and styling of a Vue component in a single file. Here's an example SFC:

```
1  <template>
2    <div>
3      <h1>Hello world</h1>
4    </div>
5  </template>
6
7  <script>
8    2 usages
9    export default {
10      name: "App"
11    }
12  </script>
13
14  <style>
15  </style>
```

# Data and methods

```
1 <template>
2   <div class="counter">
3     <div class="counter__value">{{ counter }}</div>
4     <div class="counter__actions">
5       <button @click="increment">Increment</button>
6       <button @click="decrement">Decrement</button>
7     </div>
8   </div>
9 </template>
10
11 <script>
12   2 usages
13   export default {
14     name: "TheCounter",
15     data() {
16       return {
17         counter: 0
18       }
19     },
20     methods: {
21       increment() {
22         this.counter++;
23       },
24       decrement() {
25         this.counter--;
26       }
27     }
28   }
</script>
```

# Instance Lifecycle Hooks



# Template syntax

- Text

```
<div class="counter__value">{{ counter }}</div>
```

- Raw HTML

```
<h1 v-html="msg"></h1>
```

- více si ukážeme v příkladech

# Computed Properties and Watchers

- Computed

```
computed: {  
  doubled() {  
    return this.counter * 2;  
  }  
},  
You, Moments ago • Uncommitt
```

- Watchers

```
watch: {  
  counter(value, oldValue) {  
    console.log(`counter value changed from ${oldValue} to ${value}`);  
  }  
},
```

# Class and Style Bindings

```
<template>
  <div class="counter" :class="classDefs" :style="styleDefs">
    <div v-html=msg></div>
    <div class="counter__value">{{ counter }}</div>
    <h1>Doubled</h1>
    <div class="counter__value">{{ doubled }}</div>
    <div class="counter__actions">
      <button @click="increment">Increment</button>
      <button @click="decrement">Decrement</button>
    </div>
  </div>
</template>

<script>
  2 usages
  export default {
    name: "TheCounter",
    data() {
      return {
        counter: 0,
        msg: "<h1>Counter</h1>"
      }
    },
    computed: {
      doubled() {
        return this.counter * 2;
      },
      classDefs() {
        return {
          'counter--danger': this.counter < 0
        }
      },
      styleDefs() {
        return this.counter > 5 ?
          {
            backgroundColor: "lightgreen"
          } :
          {}
      }
    },
  },
}
```

# Conditional Rendering

```
<div class="counter__actions">  
  <button v-show="counter < 10" @click="increment">Increment</button>  
  <button v-if="counter > -10" @click="decrement">Decrement</button>  
</div>| Ptacek, 33 minutes ago • init
```

# List Rendering

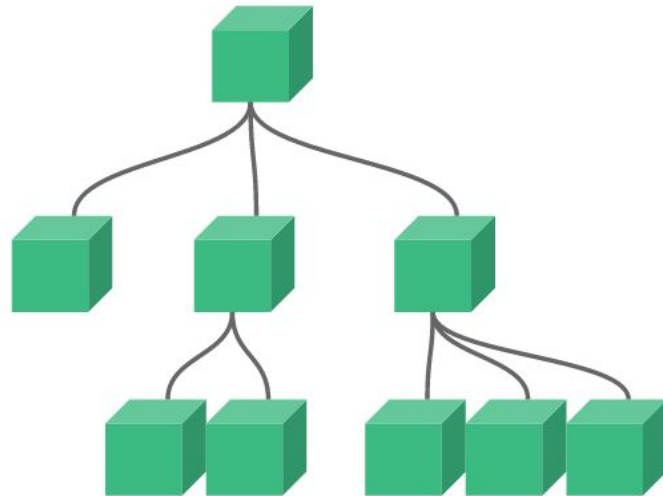
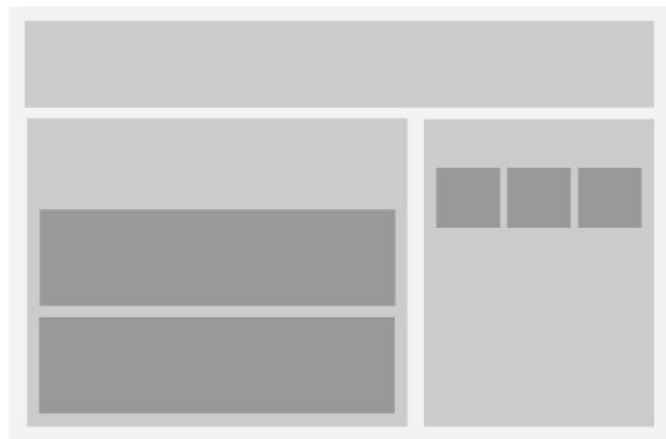
```
<div class="layout">  
|   <TheCounter v-for="i in 10" :key="i"/>  
</div>
```



# Event Handling

```
<button v-show="counter < 10" @click="increment">Increment</button>  
<button v-if="counter > -10" @click="decrement">Decrement</button>
```

# Component Organization



# Component Registration

- Global registration

```
Vue.component( id: "TheCounter", Counter);
```

- Per component

```
👤 Petr Ptacek
import TheCounter from "@components/Counter.vue";

2 usages
export default {
  name: "App", | Ptacek, 48 minutes ago • init
  components: {
    TheCounter
  }
}
</script>
```

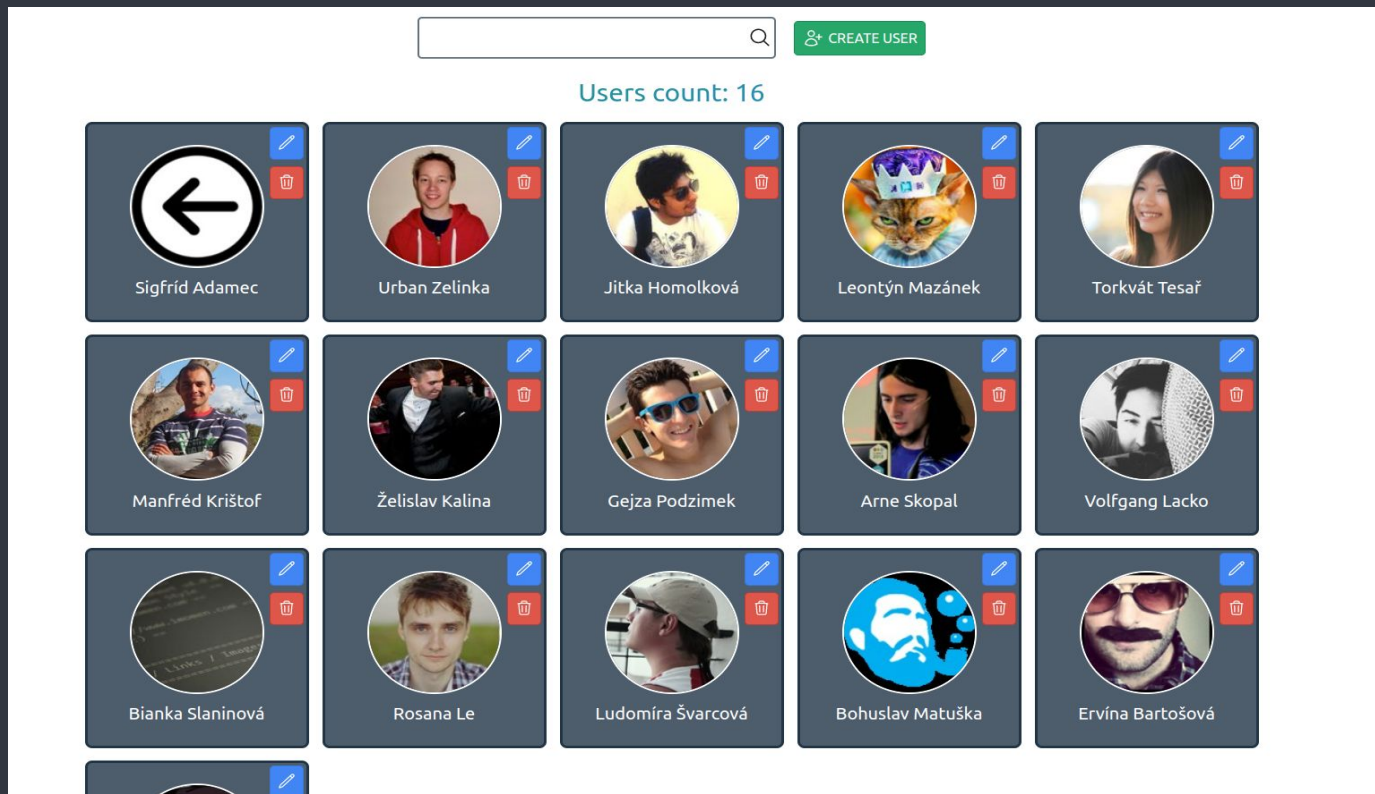
# Props

```
props: {  
  counterValue: {  
    type: Number,  
    required: true,  
    default: 0,  
    validator(val) {  
      return val >= 0;  
    }  
  }  
},
```

```
<TheCounter :counter-value="i" v-for="i in 10" :key="i"/>
```

# Ukázková aplikace rozbor

<https://github.com/petr-ptacek/olc-workshop-vue2.git>



# Užitečné odkazy

- <https://v2.vuejs.org/v2/guide/>
- <https://v2.vuejs.org/v2/style-guide/>
- <https://router.vuejs.org/>
- <https://vuex.vuejs.org/>
- <https://vueschool.io/>
- Google search