

Vue.js

Vue.js (commonly referred to as **Vue**; pronounced /vʒuː/, like "view"^[4]) is an open-source model–view–viewmodel front end JavaScript framework for building user interfaces and single-page applications.^[11] It was created by Evan You, and is maintained by him and the rest of the active core team members.^[12]

Contents

Overview

History

Versions

Features

Components

Templates

Reactivity

Transitions

Routing

Ecosystem

Official tooling

Official libraries

See also

Sources

References

External links

Overview

Vue.js features an incrementally adaptable architecture that focuses on declarative rendering and component composition. The core library is focused on the view layer only.^[13] Advanced features required for complex applications such as routing, state management and build tooling are offered via officially maintained supporting libraries and packages^[14], with Nuxt.js as one of the most popular solutions

Vue.js lets you extend HTML with HTML attributes called directives.^[15] The directives offer functionality to HTML applications, and come as either built-in or user defined directives.

History

Vue.js



<u>Original author(s)</u>	<u>Evan You</u>
<u>Initial release</u>	February 2014 ^[1]
<u>Stable release</u>	3.0.4 / December 2, 2020 ^[2]
<u>Repository</u>	<u>Vue.js</u> <u>Repository</u> (https://github.com/vuejs/vue)
<u>Written in</u>	<u>JavaScript</u> , <u>TypeScript</u> (Vue 3.0)
<u>Size</u>	33.30KB min+gzip
<u>Type</u>	<u>JavaScript</u> <u>framework</u>
<u>License</u>	<u>MIT License</u> ^[3]
<u>Website</u>	<u>vuejs.org</u> (https://vuejs.org)

Vue was created by **Evan You** after working for Google using AngularJS in a number of projects. He later summed up his thought process: "I figured, what if I could just extract the part that I really liked about Angular and build something really lightweight."^[16] The first source code commit to the project was dated July 2013, and Vue was first released the following February, in 2014.

Versions

Version	Release date	Title
3.0	September 18, 2020	<u>One Piece</u> ^[17]
2.6	February 4, 2019	<u>Macross</u> ^[18]
2.5	October 13, 2017	<u>Level E</u> ^[19]
2.4	July 13, 2017	<u>Kill la Kill</u> ^[20]
2.3	April 27, 2017	<u>JoJo's Bizarre Adventure</u> ^[21]
2.2	February 26, 2017	<u>Initial D</u> ^[22]
2.1	November 22, 2016	<u>Hunter X Hunter</u> ^[23]
2.0	September 30, 2016	<u>Ghost in the Shell</u> ^[24]
1.0	October 27, 2015	<u>Evangelion</u> ^[25]
0.12	June 12, 2015	<u>Dragon Ball</u> ^[26]
0.11	November 7, 2014	<u>Cowboy Bebop</u> ^[27]
0.10	March 23, 2014	<u>Blade Runner</u> ^[28]
0.9	February 25, 2014	<u>Animatrix</u> ^[29]
0.8	January 27, 2014	N/A ^[30]
0.7	December 24, 2013	N/A ^[31]
0.6	December 8, 2013	VueJS ^[32]

Features

Components

Vue components extend basic HTML elements to encapsulate reusable code. At a high level, components are custom elements to which the Vue's compiler attaches behavior. In Vue, a component is essentially a Vue instance with pre-defined options.^[33] The code snippet below contains an example of a Vue component. The component presents a button and prints the number of times the button is clicked:

```
<template>
  <div id="tuto">
    <button-clicked v-bind:initial-count="0"></button-clicked>
  </div>
</template>

<script>
Vue.component('button-clicked', {
  props: ['initialCount'],
  data: () => ({
    count: 0,
```

```

    }},
    template: '<button v-on:click="onClick">Clicked {{ count }} times</button>',
    computed: {
      countTimesTwo() {
        return this.count * 2;
      }
    },
    watch: {
      count(newValue, oldValue) {
        console.log(`The value of count is changed from ${oldValue} to ${newValue}.`);
      }
    },
    methods: {
      onClick() {
        this.count += 1;
      }
    },
    mounted() {
      this.count = this.initialCount;
    }
  });
});

new Vue({
  el: '#tuto',
});
</script>

```

Templates

Vue uses an [HTML](#)-based template syntax that allows binding the rendered [DOM](#) to the underlying Vue instance's data. All Vue templates are valid HTML that can be parsed by specification-compliant browsers and [HTML parsers](#). Vue compiles the templates into virtual DOM render functions. A virtual Document Object Model (or “DOM”) allows Vue to render components in its memory before updating the browser. Combined with the reactivity system, Vue is able to calculate the minimal number of components to re-render and apply the minimal amount of DOM manipulations when the app state changes.

Vue users can use template syntax or choose to directly write render functions using [JSX](#).^[34] Render functions allow application to be built from [software components](#).^[35]

Reactivity

Vue features a reactivity system that uses plain [JavaScript](#) objects and optimized re-rendering. Each component keeps track of its reactive dependencies during its render, so the system knows precisely when to re-render, and which components to re-render.^[36]

Transitions

Vue provides a variety of ways to apply transition effects when items are inserted, updated, or removed from the [DOM](#). This includes tools to:

- Automatically apply classes for [CSS](#) transitions and animations
- Integrate third-party CSS animation libraries, such as [Animate.css](#)
- Use JavaScript to directly manipulate the DOM during transition hooks
- Integrate third-party JavaScript animation libraries, such as [Velocity.js](#)

When an element wrapped in a transition component is inserted or removed, this is what happens:

1. Vue will automatically sniff whether the target element has CSS transitions or animations applied. If it does, CSS transition classes will be added/removed at appropriate timings.
2. If the transition component provided JavaScript hooks, these hooks will be called at appropriate timings.
3. If no CSS transitions/animations are detected and no JavaScript hooks are provided, the DOM operations for insertion and/or removal will be executed immediately on next frame.^{[37][38]}

Routing

A traditional disadvantage of single-page applications (SPAs) is the inability to share links to the exact "sub" page within a specific web page. Because SPAs serve their users only one URL-based response from the server (it typically serves `index.html` or `index.vue`), bookmarking certain screens or sharing links to specific sections is normally difficult if not impossible. To solve this problem, many client-side routers delimit their dynamic URLs with a "hashbang" (`#!`), e.g. `page.com/#!/`. However, with HTML5 most modern browsers support routing without hashbangs.

Vue provides an interface to change what is displayed on the page based on the current URL path – regardless of how it was changed (whether by emailed link, refresh, or in-page links). Additionally, using a front-end router allows for the intentional transition of the browser path when certain browser events (i.e. clicks) occur on buttons or links. Vue itself doesn't come with front-end hashed routing. But the open source "vue-router" package provides an API to update the application's URL, supports the back button (navigating history), and email password resets or email verification links with authentication URL parameters. It supports mapping nested routes to nested components and offers fine-grained transition control. With Vue, developers are already composing applications with small building blocks building larger components. With vue-router added to the mix, components must merely be mapped to the routes they belong to, and parent/root routes must indicate where children should render.^[39]

```
<div id="app">
  <router-view></router-view>
</div>
...

<script>
...
const User = {
  template: '<div>User {{ $route.params.id }}</div>'
};

const router = new VueRouter({
  routes: [
    { path: '/user/:id', component: User }
  ]
});
...
</script>
```

The code above:

1. Sets a front-end route at `websitename.com/user/<id>`.
2. Which will render in the User component defined in `(const User...)`
3. Allows the User component to pass in the particular id of the user which was typed into the URL using the `$route` object's params key: `$route.params.id`.
4. This template (varying by the params passed into the router) will be rendered into `<router-view></router-view>` inside the DOM's `div#app`.
5. The finally generated HTML for someone typing in: `websitename.com/user/1` will be:

```
<div id="app">
  <div>
    <div>User 1</div>
  </div>
</div>
```

[40]

Ecosystem

The core library comes with tools and libraries both developed by the core team and contributors.

Official tooling

- **Devtools** – Browser devtools extension for debugging Vue.js applications
- **Vue CLI** – Standard Tooling for rapid Vue.js development
- **Vue Loader** – a webpack loader that allows the writing of Vue components in a format called Single-File Components (SFCs)


Official libraries

- **Vue Router** – The official router for Vue.js
- **Vuex** – Flux-inspired Centralized State Management for Vue.js
- **Vue Server Renderer** – Server-Side Rendering for Vue.js

See also

- [Comparison of JavaScript frameworks](#)
- [React](#)
- [AngularJS](#)
- [Angular](#)
- [Quasar Framework](#)
- [JavaScript framework](#)
- [JavaScript library](#)
- [Model–view–ViewModel](#)
- [Nuxt.js](#)

Sources

 This article incorporates text from a [free content](#) work. Licensed under [MIT License](#) [License statement/permission](#) on Wikimedia Commons (<https://github.com/vuejs/vuejs.org/blob/fdc66945e18020cdbdc729fd7ab8005d123216a7/LICENSE>). Text taken from [Vue.js Guide](#) (<https://vuejs.org/v2/guide>), Vue.js,


References

1. "First Week of Launching Vue.js" (<http://blog.evanyou.me/2014/02/11/first-week-of-launching-a-n-oss-project/>). *Evan You*.

2. "vue-next/CHANGELOG.md" (<https://github.com/vuejs/vue-next/blob/master/CHANGELOG.md>). *GitHub*. Retrieved December 3, 2020.
3. "vue/LICENSE" (<https://github.com/vuejs/vue/blob/dev/LICENSE>). *GitHub*. Retrieved April 17, 2017.
4. "Guide: What is Vue.js?" (<https://vuejs.org/v2/guide/index.html#What-is-Vue-js>). *Vue.js*. Retrieved January 3, 2020.
5. Macrae, Callum (2018). *Vue.js: Up and Running: Building Accessible and Performant Web Apps* (<https://books.google.com/books?id=bJpNDwAAQBAJ>). O'Reilly Media. ISBN 9781491997215.
6. Nelson, Brett (2018). *Getting to Know Vue.js: Learn to Build Single Page Applications in Vue from Scratch* (<https://books.google.com/books?id=mTpsDwAAQBAJ>). Apress. ISBN 9781484237816.
7. Yerburgh, Edd (2019). *Testing Vue.js Applications* (<https://books.google.com/books?id=7-FAtAEACAAJ>). Manning Publications. ISBN 9781617295249.
8. Freeman, Adam (2018). *Pro Vue.js 2* (<https://books.google.com/books?id=HQFuDwAAQBAJ>). Apress. ISBN 9781484238059.
9. Franklin, Jack; Wanyoike, Michael; Bouchefra, Ahmed; Silas, Kingsley; Campbell, Chad A.; Jacques, Nilson; Omole, Olayinka; Mulders, Michiel (2019). *Working with Vue.js* (<https://books.google.com/books?id=OKScDwAAQBAJ>). SitePoint. ISBN 9781492071440.
10. "Introduction — Vue.js" (<https://vuejs.org/v2/guide/#What-is-Vue-js>). Retrieved March 11, 2017.
11. [5][6][7][8][9][10]
12. "Meet the Team — Vue.js" (<https://vuejs.org/v2/guide/team.html>). *vuejs.org*. Retrieved September 23, 2019.
13. "Introduction — Vue.js" (<https://vuejs.org/v2/guide/>). *vuejs.org*. Retrieved May 27, 2020.
14. "Evan is creating Vue.js | Patreon" (<https://www.patreon.com/evanyou>). *Patreon*. Retrieved March 11, 2017.
15. "What is Vue.js" (https://www.w3schools.com/whatis/whatis_vue.asp). *www.w3schools.com*. Retrieved January 21, 2020.
16. "Between the Wires | Evan You" (<https://web.archive.org/web/20170603052649/https://betweenthewires.org/2016/11/03/evan-you/>). *Between the Wires*. November 3, 2016. Archived from the original (<https://betweenthewires.org/2016/11/03/evan-you/>) on June 3, 2017. Retrieved August 26, 2017.
17. "v3.0.0 One Piece" (<https://github.com/vuejs/vue-next/releases/tag/v3.0.0>). *GitHub*. September 18, 2020. Retrieved September 23, 2020.
18. "v2.6.0 Macross" (<https://github.com/vuejs/vue/releases/tag/v2.6.0>). *GitHub*. February 4, 2019. Retrieved September 23, 2020.
19. "v2.5.0 Level E" (<https://github.com/vuejs/vue/releases/tag/v2.5.0>). *GitHub*. October 13, 2017. Retrieved September 23, 2020.
20. "v2.4.0 Kill la Kill" (<https://github.com/vuejs/vue/releases/tag/v2.4.0>). *GitHub*. July 13, 2017. Retrieved September 23, 2020.
21. "v2.3.0 JoJo's Bizarre Adventure" (<https://github.com/vuejs/vue/releases/tag/v2.3.0>). *GitHub*. April 27, 2017. Retrieved September 23, 2020.
22. "v2.2.0 Initial D" (<https://github.com/vuejs/vue/releases/tag/v2.2.0>). *GitHub*. February 26, 2017. Retrieved September 23, 2020.
23. "v2.1.0 Hunter X Hunter" (<https://github.com/vuejs/vue/releases/tag/v2.1.0>). *GitHub*. November 22, 2016. Retrieved September 23, 2020.
24. "v2.0.0 Ghost in the Shell" (<https://github.com/vuejs/vue/releases/tag/v2.0.0>). *GitHub*. September 30, 2016. Retrieved September 23, 2020.
25. "1.0.0 Evangelion" (<https://github.com/vuejs/vue/releases/tag/1.0.0>). *GitHub*. October 27, 2015. Retrieved September 23, 2020.

26. "0.12.0: Dragon Ball" (<https://github.com/vuejs/vue/releases/tag/0.12.0>). GitHub. June 12, 2015. Retrieved September 23, 2020.
27. "v0.11.0: Cowboy Bebop" (<https://github.com/vuejs/vue/releases/tag/0.11.0>). GitHub. November 7, 2014. Retrieved September 23, 2020.
28. "v0.10.0: Blade Runner" (<https://github.com/vuejs/vue/releases/tag/v0.10.0>). GitHub. March 23, 2014. Retrieved September 23, 2020.
29. "v0.9.0: Animatrix" (<https://github.com/vuejs/vue/releases/tag/v0.9.0>). GitHub. February 25, 2014. Retrieved September 23, 2020.
30. "v0.8.0" (<https://github.com/vuejs/vue/releases/tag/v0.8.0>). GitHub. January 27, 2014. Retrieved September 23, 2020.
31. "v0.7.0" (<https://github.com/vuejs/vue/releases/tag/v0.7.0>). GitHub. December 24, 2013. Retrieved September 23, 2020.
32. "0.6.0: VueJS" (<https://github.com/vuejs/vue/releases/tag/0.6.0>). GitHub. December 8, 2013. Retrieved September 23, 2020.
33. "Components — Vue.js" (<https://vuejs.org/v2/guide/components.html>). Retrieved March 11, 2017.
34. "Template Syntax — Vue.js" (<https://vuejs.org/v2/guide/syntax.html>). Retrieved March 11, 2017.
35. "Vue 2.0 is Here!" (<https://medium.com/the-vue-point/vue-2-0-is-here-ef1f26acf4b8#77d9>). *The Vue Point*. September 30, 2016. Retrieved March 11, 2017.
36. "Reactivity in Depth — Vue.js" (<https://vuejs.org/v2/guide/reactivity.html>). Retrieved March 11, 2017.
37. "Transition Effects — Vue.js" (<https://vuejs.org/v2/guide/transitions.html>). Retrieved March 11, 2017.
38. "Transitioning State — Vue.js" (<https://vuejs.org/v2/guide/transitioning-state.html>). Retrieved March 11, 2017.
39. "Routing — Vue.js" (<https://vuejs.org/v2/guide/routing.html>). Retrieved March 11, 2017.
40. You, Evan. "Vue Nested Routing (2)" (<https://router.vuejs.org/en/essentials/nested-routes.html>). *Vue Home Page (subpage)*. Retrieved May 10, 2017.

External links

- [Official website \(https://vuejs.org/\)](https://vuejs.org/) 
-

Retrieved from "<https://en.wikipedia.org/w/index.php?title=Vue.js&oldid=993112659>"

This page was last edited on 8 December 2020, at 21:19 (UTC).

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. By using this site, you agree to the Terms of Use and Privacy Policy. Wikipedia® is a registered trademark of the Wikimedia Foundation, Inc., a non-profit organization.