



链滴

Vue3 使用 Mitt 替代 EventBus

作者: [Suvern](#)

原文链接: <https://ld246.com/article/1609174237860>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



前言

前段时间接触了Vue，作为一个合格——————的前端，流行的框架还是得会点的

不过一直听说Vue开箱即用，**VueX**、**Vue Router**、**Event Bus**一堆现成的玩意，然而接触了Vue3才发现，EventBus被去掉了???

这怎么可以（暂时不提 **Composition API**，我菜鸡还没学会。。。）？

Mitt

Mitt是一个体积极小的第三方消息发布/订阅式JavaScript库

官方文档是与框架无关的，所以这个玩意其实React、Vue都可以用

本文以Vue3为例，介绍Mitt库的导入和使用

准备

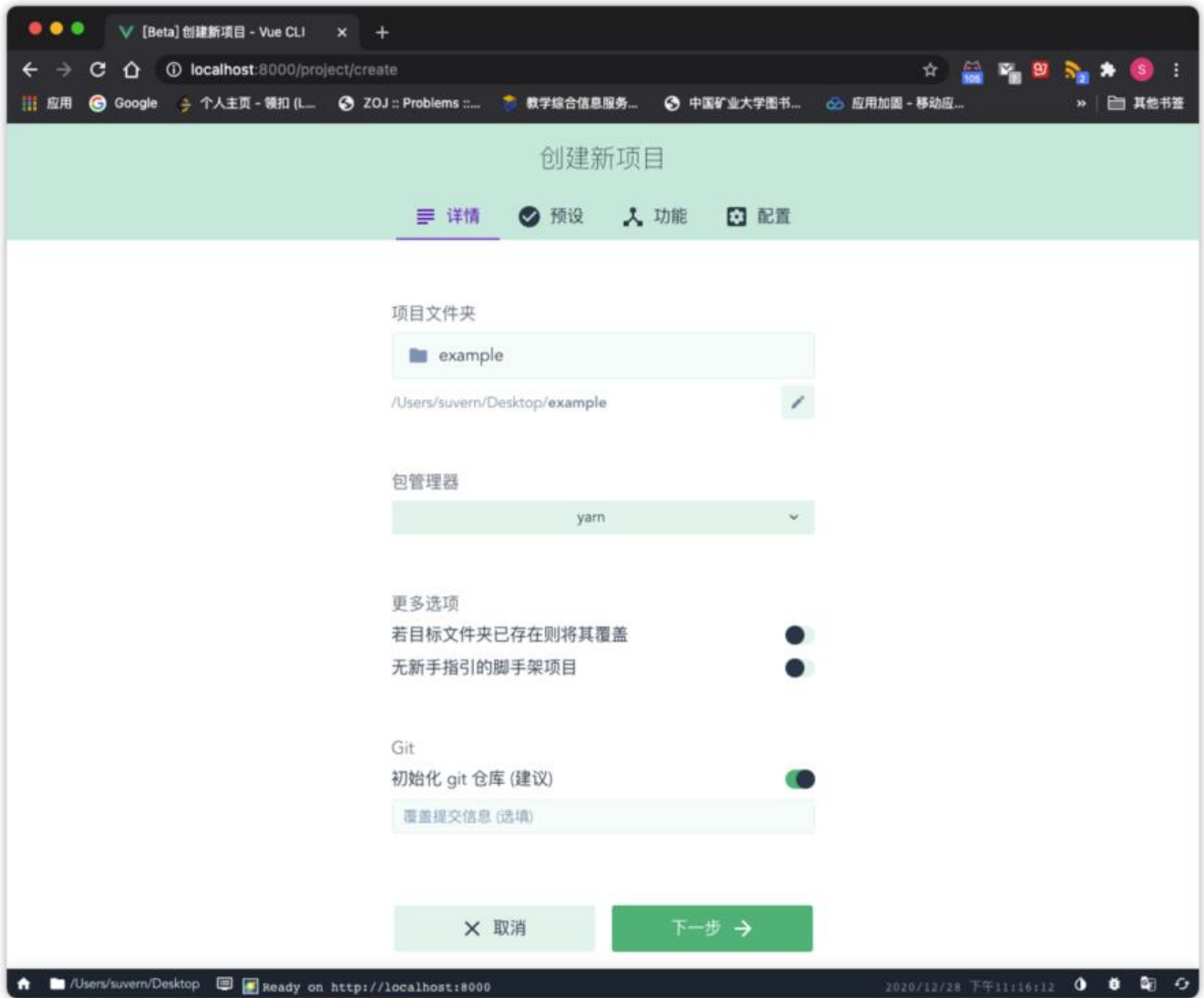
利用vue脚手架创建一个vue3项目，不使用TypeScript

在这里，我比较喜欢用

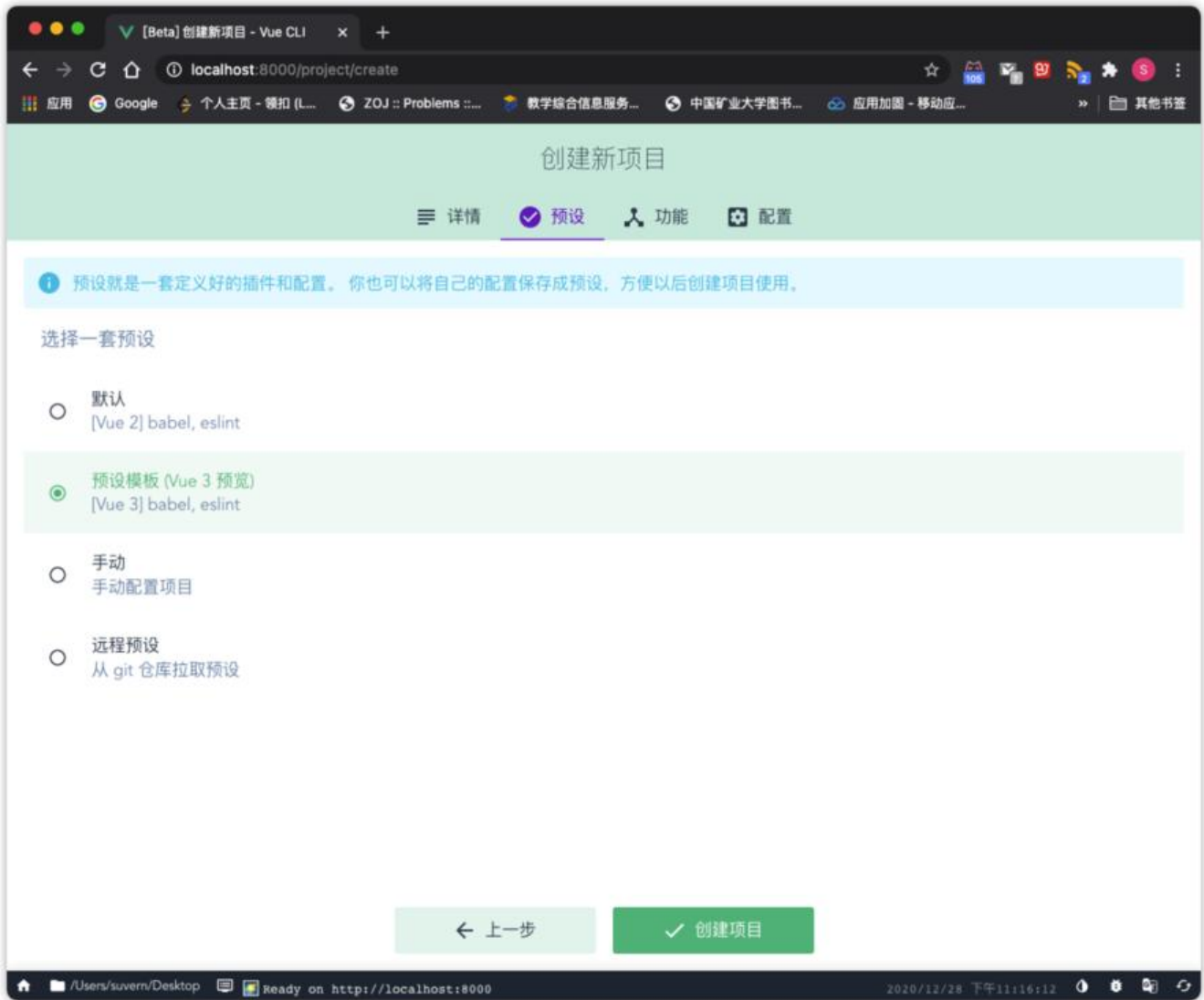
```
$ vue ui
```

命令，打开一个图形化页面来创建项目

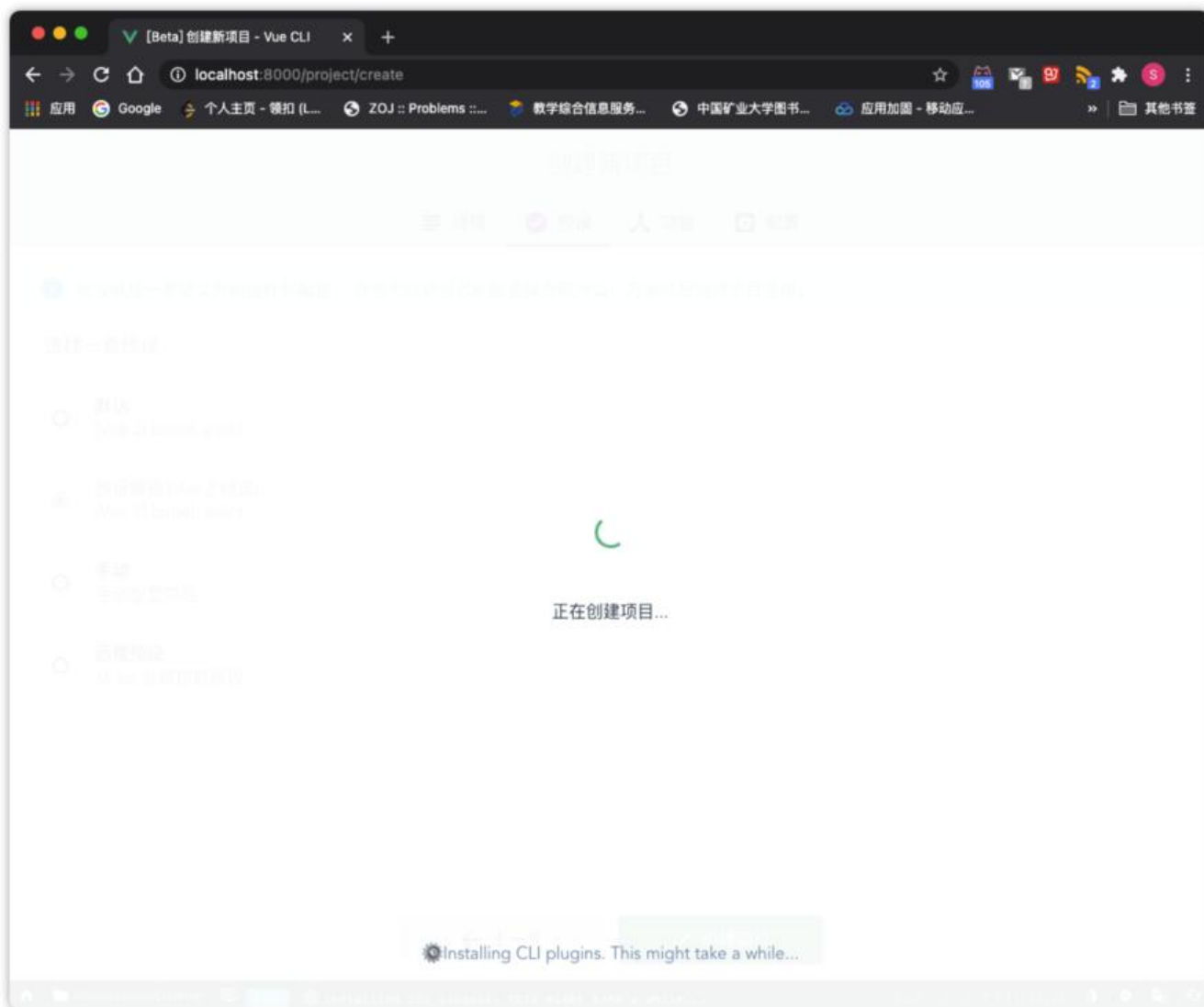
在这里，项目命名为 **example**



选择Vue3模板后创建项目



稍等几分钟后，创建完成，用编辑器或IDE打开项目文件夹



这样一个Vue3的项目就创建好了

安装

在项目的根目录下执行：

```
$ npm i mitt --save
```

导入

1.

在 `src/` 下新建一个 `bus.js` 文件，内容如下：

```
/* bus.js */  
  
import mitt from "mitt";  
  
const bus = {}  
const emitter = mitt()
```

```
bus.$on = emitter.on
bus.$off = emitter.off
bus.$emit = emitter.emit
```

```
export default bus
```

这段代码引入了mitt，并利用mitt()构造器生成了一个emit实例，同时把 on、off和 emit方法赋值给空对象 bus，作为静态方法

2.

打开 src/main.js内容如下：

```
/* main.js */

import { createApp } from 'vue'
import App from './App.vue'

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

改造 main.js，导入刚才的 bus对象，并挂载到全局

改造后的 main.js：

```
/* main.js */

import { createApp } from 'vue'
import App from './App.vue'
import bus from './bus'

const app = createApp(App)

app.config.globalProperties.$bus = bus

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

使用

这里，我们在父、子组件间通过刚才挂载的全局对象 \$bus来进行事件传递，并使用Vue3新增的 reactive来实现一个响应式组件

改造

改造默认模板里的 src/App.vue和 src/components/HelloWorld.vue，改造后的 App.vue：

```
<template>
  
  <button @click="handleClick()">这是一个按钮</button>
  <HelloWorld msg="Welcome to Your Vue.js App"/>
</template>

<script>
import HelloWorld from './components/HelloWorld.vue'
```

```
export default {
  name: 'App',
  components: {
    HelloWorld
  },
  methods: {
    handleClick() {
      this.$bus.$emit('click')
    }
  }
}
</script>
```

```
<style>
#app {
  font-family: Avenir, Helvetica, Arial, sans-serif;
  -webkit-font-smoothing: antialiased;
  -moz-osx-font-smoothing: grayscale;
  text-align: center;
  color: #2c3e50;
  margin-top: 60px;
}
</style>
```

改造后的 **HelloWorld.vue**:

```
<template>
<div class="hello">
  <h1>{{ msg }}</h1>
  <p>
    For a guide and recipes on how to configure / customize this project, <br/>
    check out the
    <a href="https://cli.vuejs.org" target="_blank" rel="noopener"
    >vue-cli documentation</a>
  >.
</p>
<h3>Installed CLI Plugins</h3>
<ul>
  <li v-for="i in links" :key="i.text">
    <a href="{{ i.link }}" target="_blank" rel="noopener">{{ i.text }}</a>
  </li>
</ul>
<h3>Essential Links</h3>
<ul>
  <li>
    <a href="https://vuejs.org" target="_blank" rel="noopener">Core Docs</a>
  </li>
  <li>
    <a href="https://forum.vuejs.org" target="_blank" rel="noopener"
    >Forum</a>
  >
</li>
<li>
```

```

    <a href="https://chat.vuejs.org" target="_blank" rel="noopener"
    >Community Chat</a
    >
  </li>
  <li>
    <a href="https://twitter.com/vuejs" target="_blank" rel="noopener"
    >Twitter</a
    >
  </li>
  <li>
    <a href="https://news.vuejs.org" target="_blank" rel="noopener">News</a>
  </li>
</ul>
<h3>Ecosystem</h3>
<ul>
  <li>
    <a href="https://router.vuejs.org" target="_blank" rel="noopener"
    >vue-router</a
    >
  </li>
  <li>
    <a href="https://vuex.vuejs.org" target="_blank" rel="noopener">vuex</a>
  </li>
  <li>
    <a
      href="https://github.com/vuejs/vue-devtools#vue-devtools"
      target="_blank"
      rel="noopener"
    >vue-devtools</a
    >
  </li>
  <li>
    <a href="https://vue-loader.vuejs.org" target="_blank" rel="noopener"
    >vue-loader</a
    >
  </li>
  <li>
    <a
      href="https://github.com/vuejs/awesome-vue"
      target="_blank"
      rel="noopener"
    >awesome-vue</a
    >
  </li>
</ul>
</div>
</template>

<script>
import {reactive} from "vue";

export default {
  name: "HelloWorld",
  props: {

```



```
    msg: String,
  },
  setup() {
    const links = reactive([
      {link: 'https://github.com/vuejs/vue-cli/tree/dev/packages/%40vue/cli-plugin-babel', text:
babel'},
      {link: 'https://github.com/vuejs/vue-cli/tree/dev/packages/%40vue/cli-plugin-eslint', text:
eslint'}
    ])
    const add = () => links.push({
      link: 'https://github.com/developit/mitt',
      text: 'mitt'
    })
    return {links, add}
  },
  mounted() {
    this.$bus.$on("click", this.add);
  },
};
</script>
```

```
<!-- Add "scoped" attribute to limit CSS to this component only -->
<style scoped>
h3 {
  margin: 40px 0 0;
}

ul {
  list-style-type: none;
  padding: 0;
}

li {
  display: inline-block;
  margin: 0 10px;
}

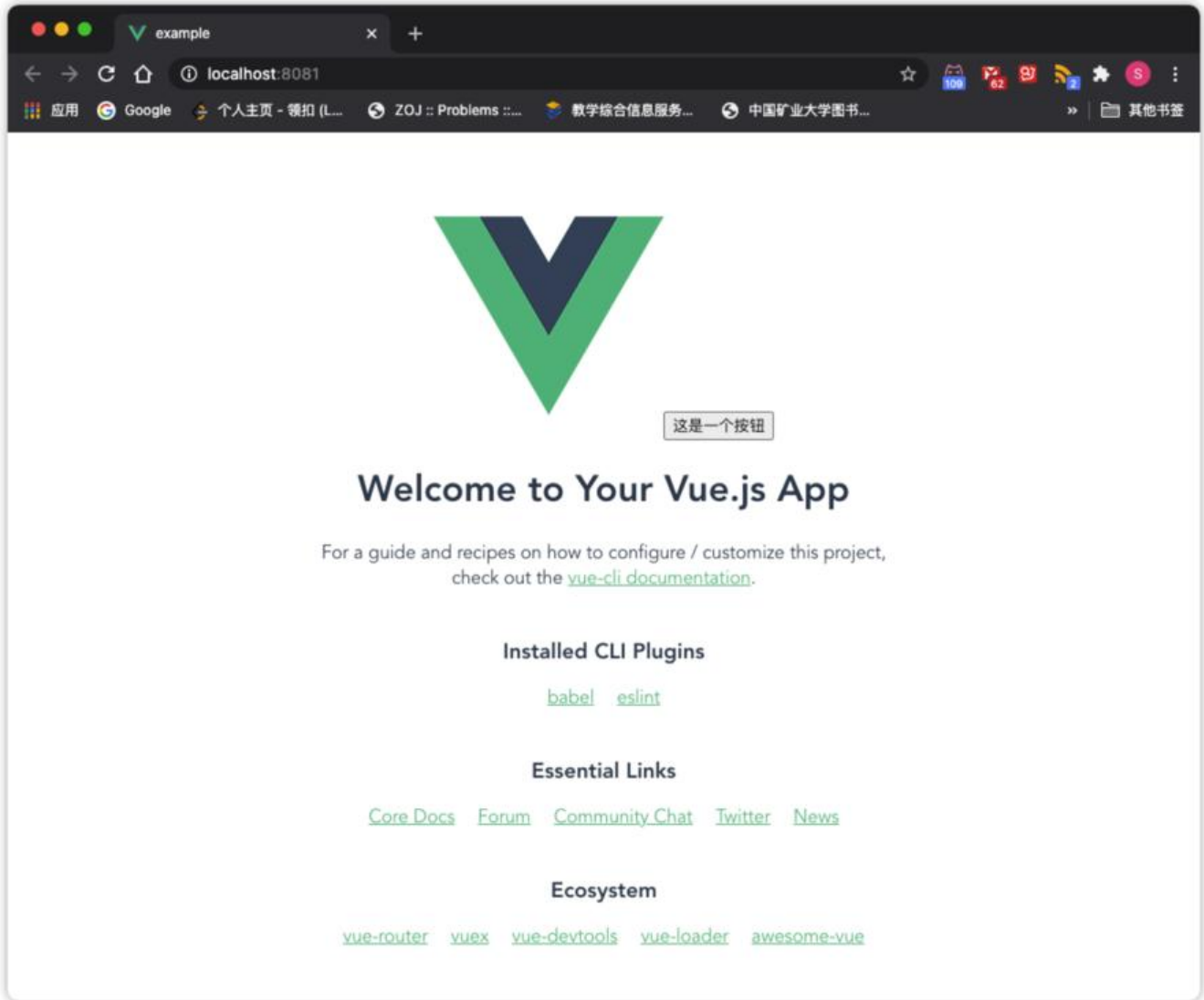
a {
  color: #42b983;
}
</style>
```

效果

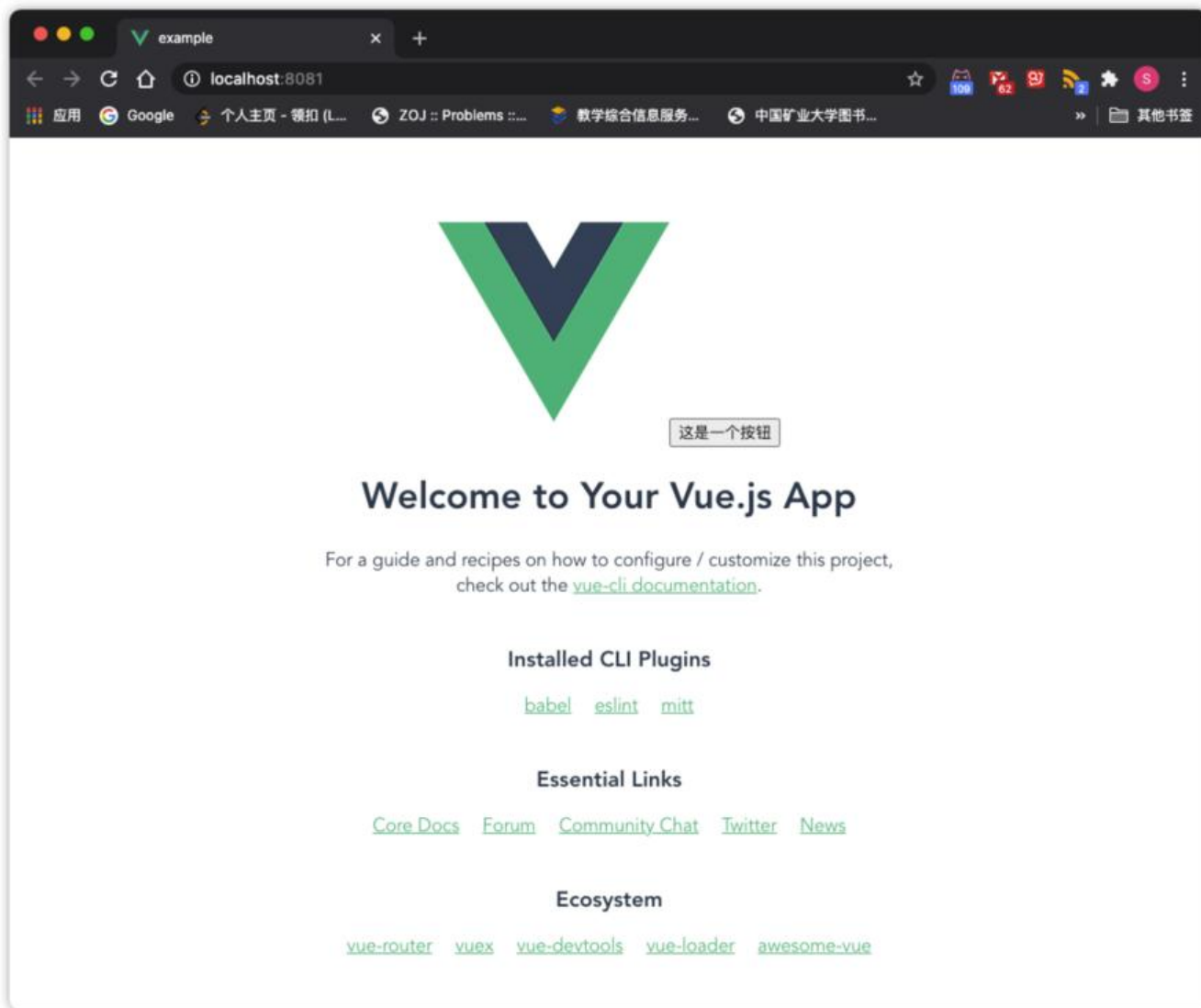
在项目根目录执行：

```
$ npm run serve
```

打开浏览器到相应的url查看页面



点击按钮，发现 “Installed CLI Plugins” 下面增加了一个 “mitt”



总结

自此，本篇文章就结束了，相关代码已上传至Github: [Suvern/vue3-mitt-example](#)提供参考

Vue3的更新，让很多程序员感慨：“学不动了”

不过我认为，敲代码的乐趣更多的在于学习和发现新东西

在编写这篇文章之前，我因为最近开发一个Vue3的项目查阅了很多资料，然而国内搜到关于 **VueX**和 **ue Router**之类的教程和文章都是使用Vue2.x，内容也参差不齐，所以怒查文档，结果还是从官方模里找到了答案

所以还是学吧☹️☹️☹️