

Workshop:

# State Management

## in Vue.js Applications

**Fill for at**  
**<https://goo.gl/zcj9W9>**

**npm install -g pnpm**

**or**

**npm install -g yarn**

```
git clone https://github.com/znck/vuex-workshop
```

**cd vuex-workshop**

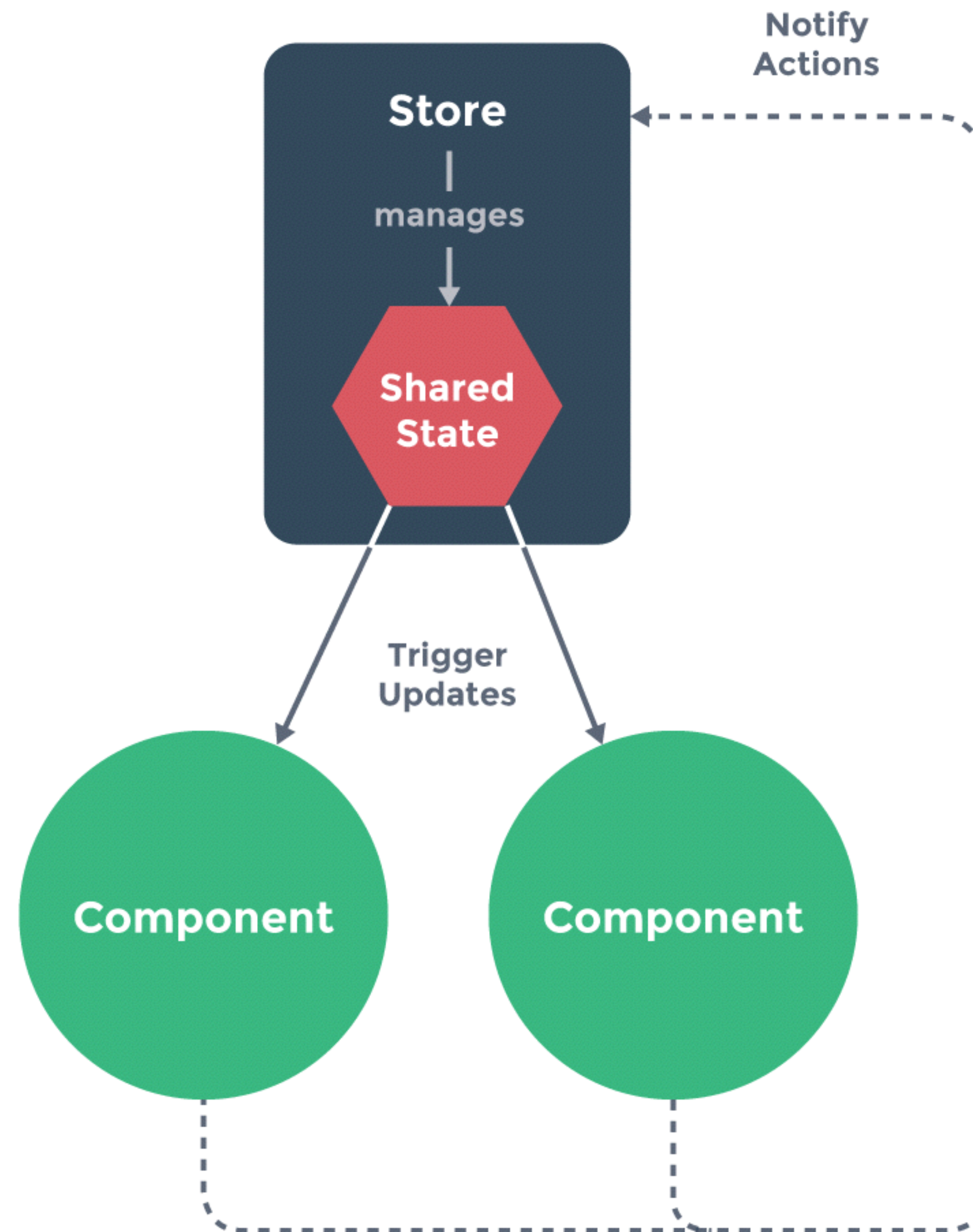
**pnpm install**

**pnpm run dev**

**Why?**

# **Common Problems In Large Frontend Apps**

- 1. Sharing state between multiple components**
- 2. Unpredictable state mutations**







## store.js

```
import Vue from 'vue'
import Vuex, { Store } from 'vuex'

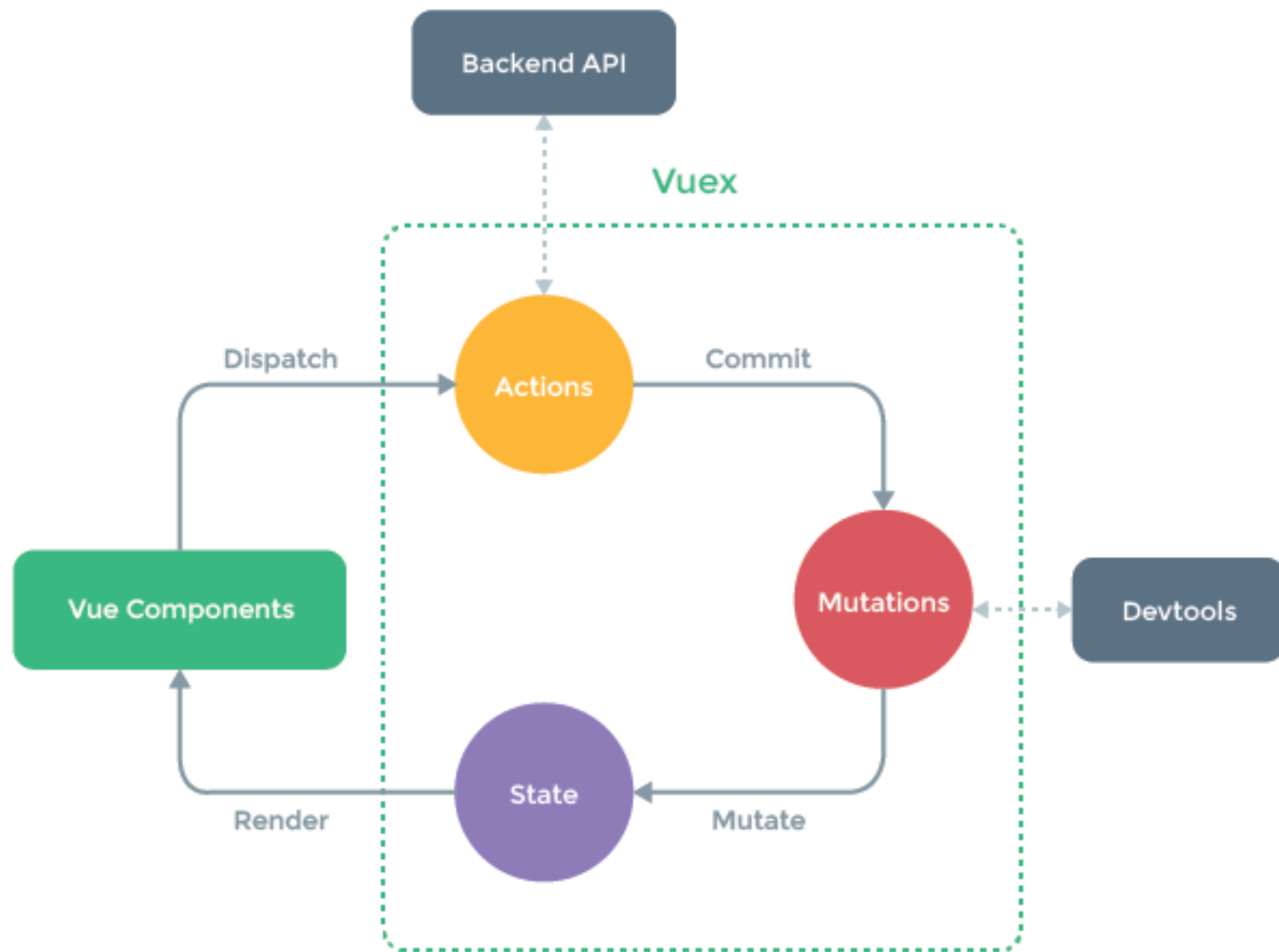
Vue.use(Vuex)

export default new Store({
  state: { cnt: 0 },
  mutations: {
    INC (state) {
      state.cnt += 1
    },
    DEC (state) {
      state.cnt -= 1
    }
  },
  actions: {
    change ({ commit }, delta) {
      delta > 0 ? commit('INC') : commit('DEC')
    }
  }
})
```

## app.js

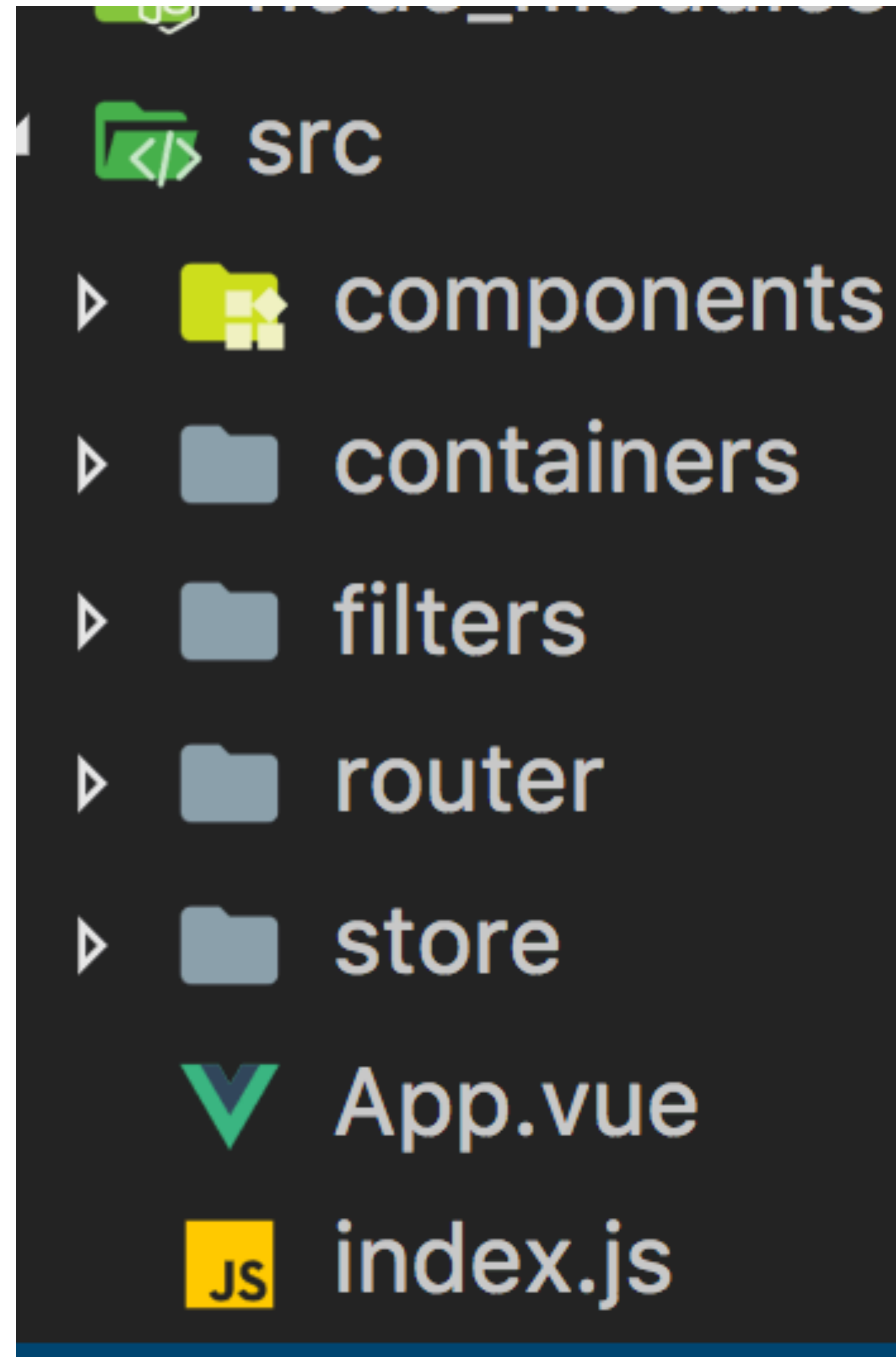
```
import { mapAction, mapState } from 'vuex'
import store from './store'

new Vue({
  el: '#app',
  template: `<div>
    count: {{ cnt }}
    <button @click="change(-1)"> - </button>
    <button @click="change(1)"> + </button>
  </div>`,
  computed: mapState(['cnt']),
  methods: mapAction(['change']),
  store
})
```



# Code Overview

# Project Directory Structure



# A Simple Store

# Consuming API

# Modules