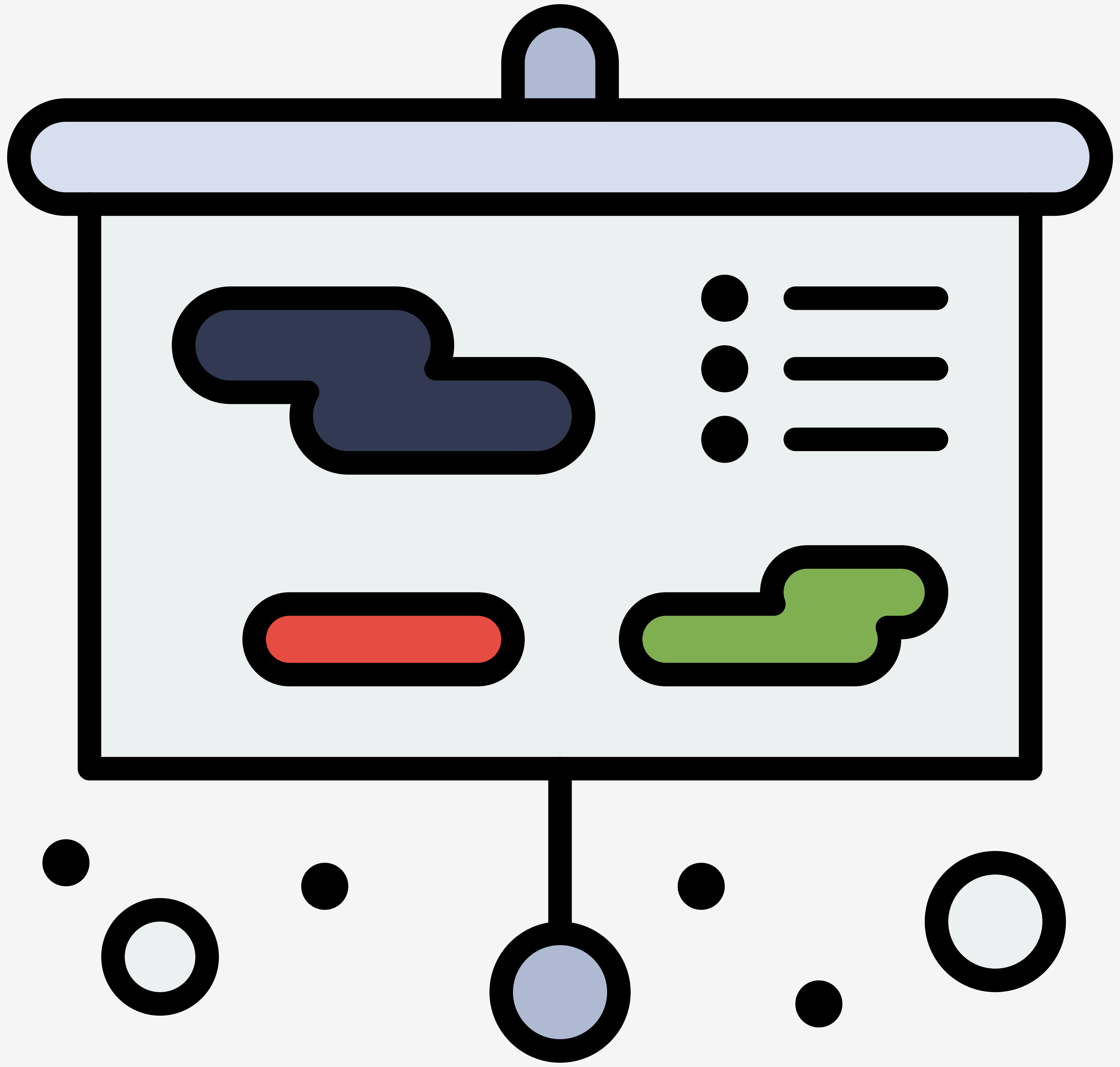
WEB PRODUCTION

Lecture 4

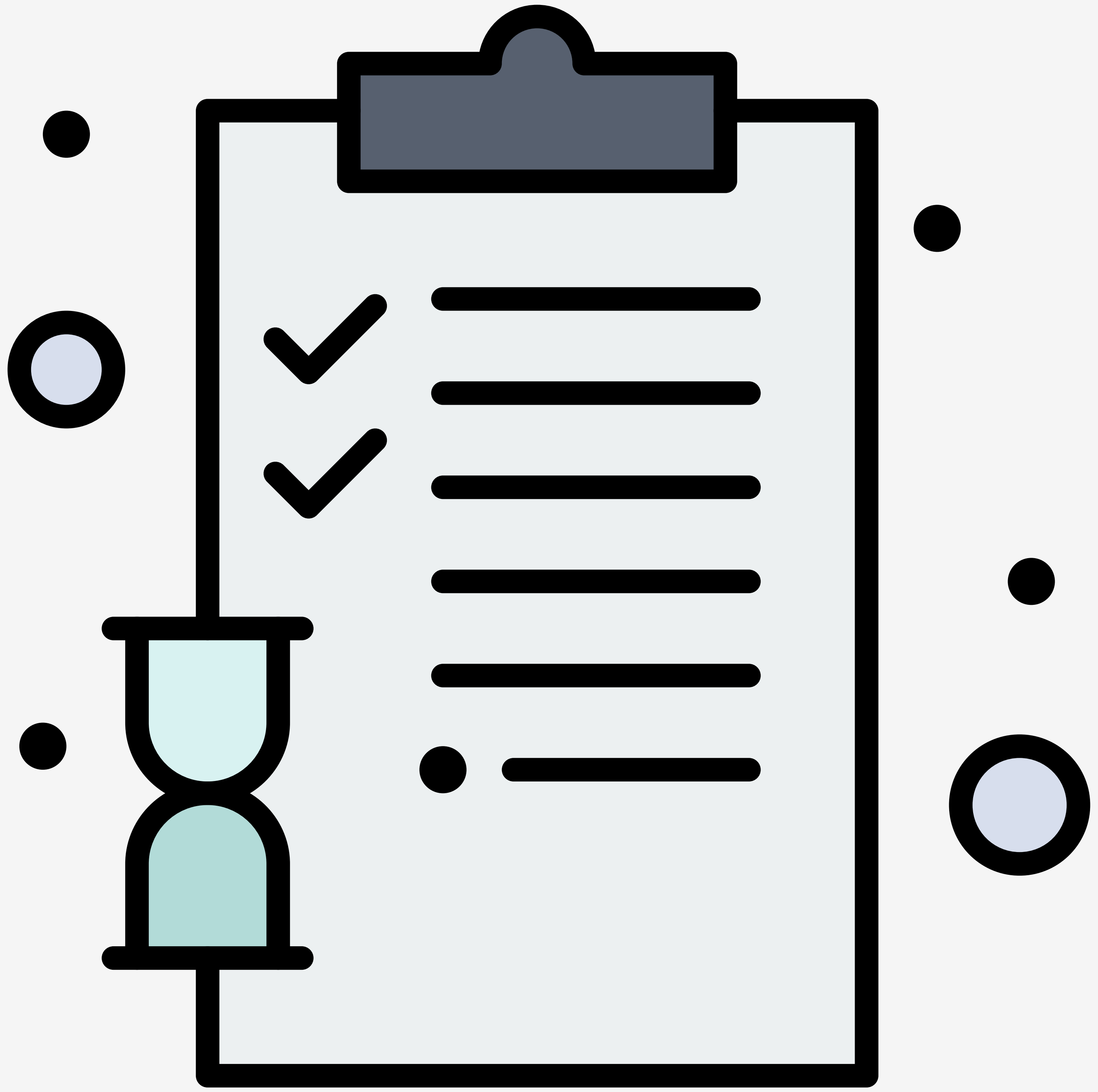
Graphics by Flatart on freepik.com

TODAY'S TOPICS



- Vue Components
- Participation: Vuetip
- Exercise: Grocery List

ANNOUNCEMENTS

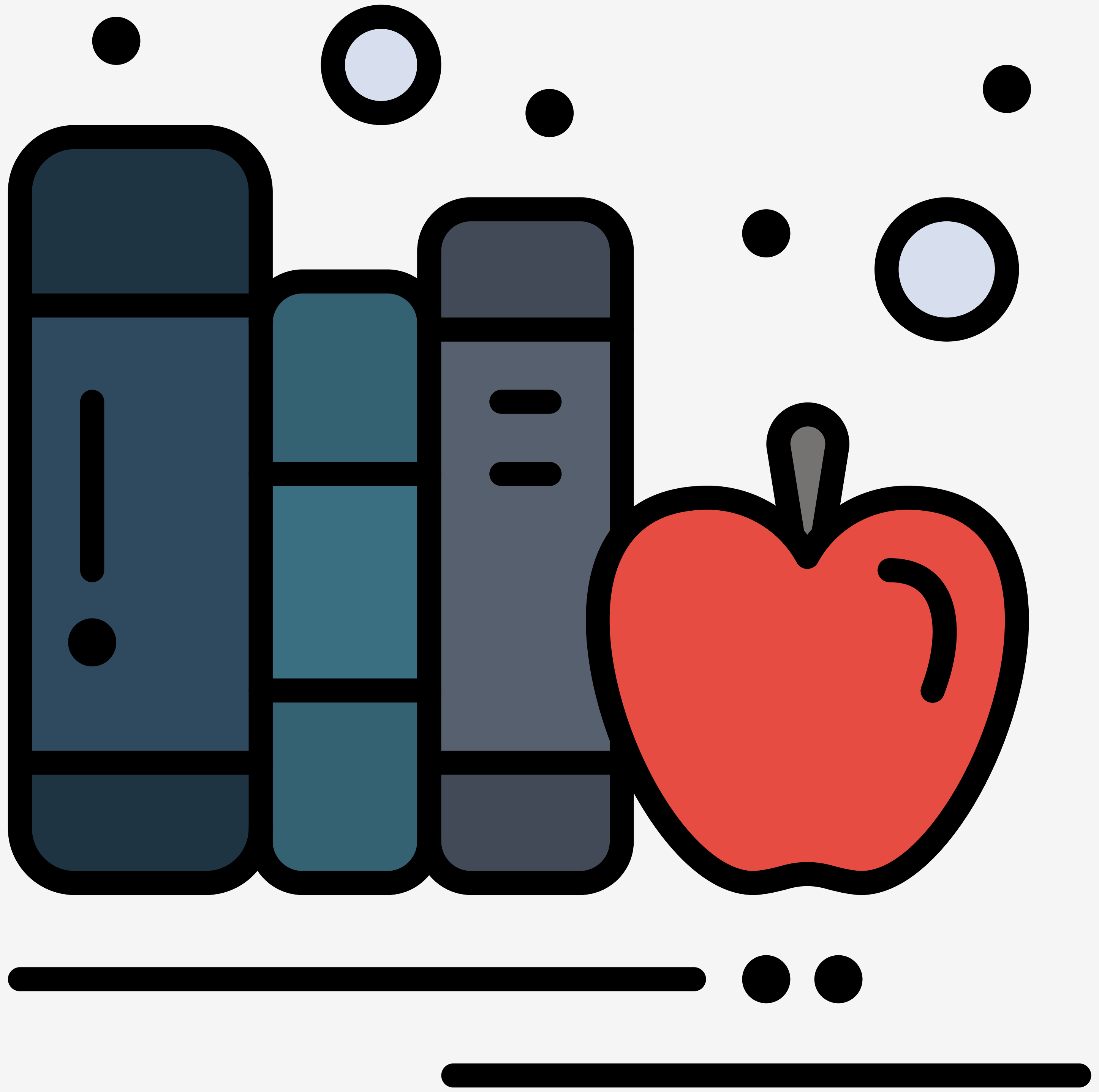


- Sign-in Sheet
- Recording
- Assignment Changes

QUESTIONS

VUE COMPONENTS

VUE COMPONENTS



- Components are reusable Vue instances
- Components have there own names, data, methods and computed properties
- Components are used by adding custom elements
- Components can be defined by directing calling the `component` method of the Vue Class

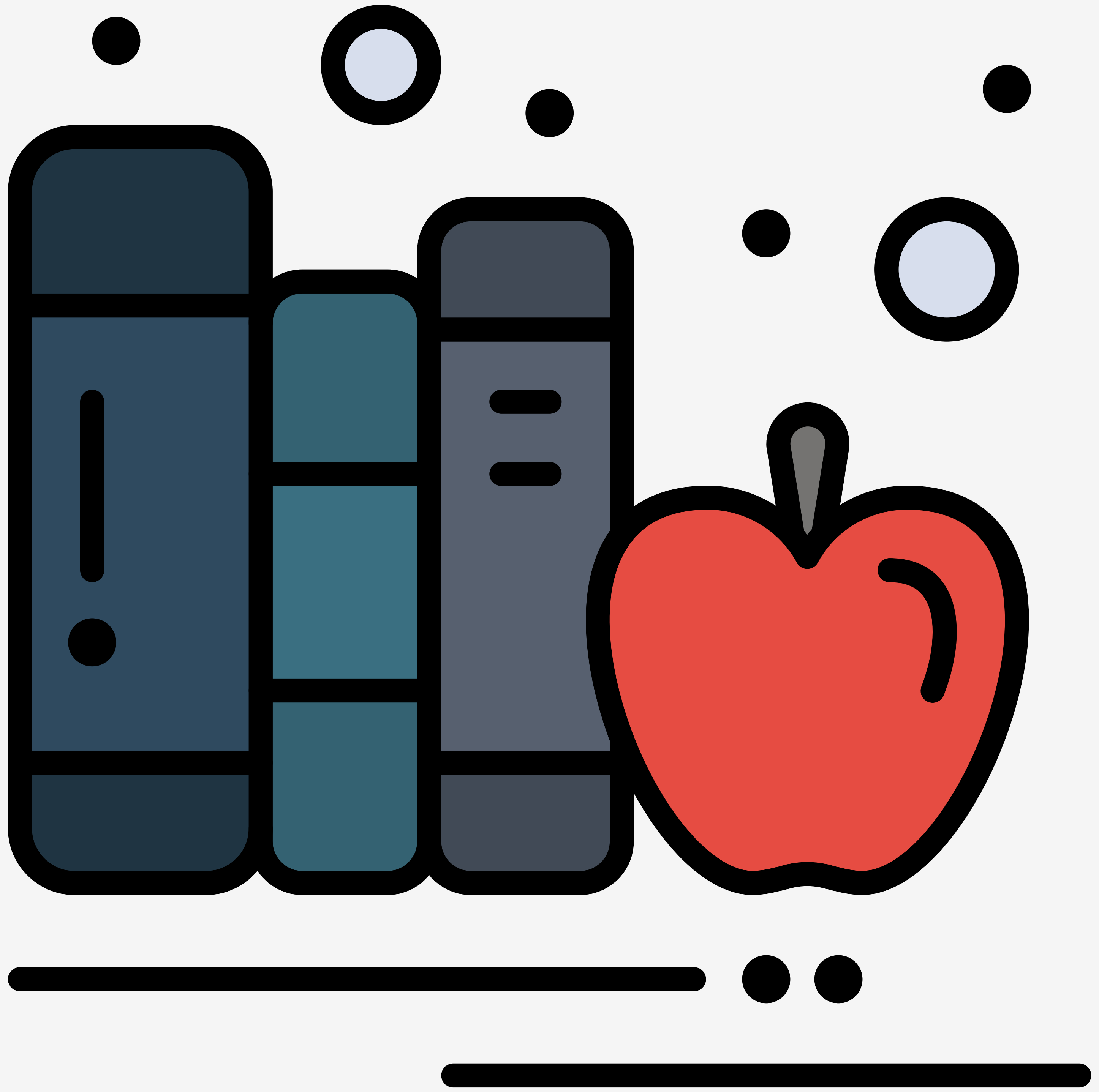
COMPONENTS

```
<div id="app">
  <simple-button></simple-button>
</div>
```

```
// Define a new component
Vue.component('simple-button', {
  template: `
  <button class="btn btn-primary">
    Click me!
  </button>`
})

new Vue({ el: '#app' })
```

VUE COMPONENTS



- When adding the `data` property to a component a function must be used
- Component Templates must have a single root element
- Component Templates can include other component elements
- Component elements can be adding to the template syntax multiple times

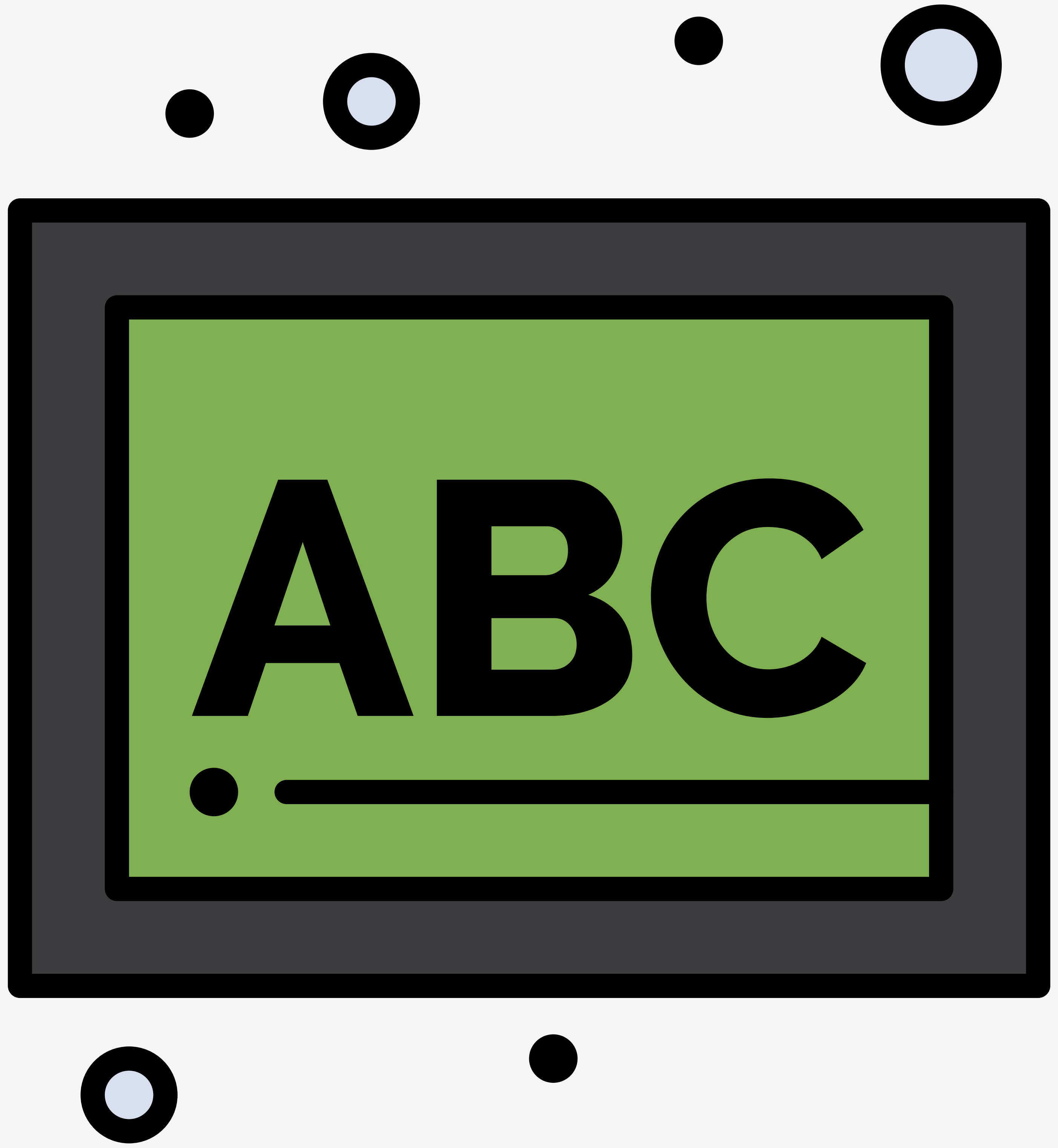
COMPONENTS

```html
<div id="app">
  <simple-button></simple-button>
  <simple-button></simple-button>
  <simple-button></simple-button>
</div>
```

```javascript
Vue.component('simple-button', {
  data: function () {
    return {
      text: 'Click Me',
      type: 'success'
    }
  },
  template: `
  <button class="btn m-1" :class="'btn-'+type">
    {{ text }}
  </button>`
})

new Vue({ el: '#app' })
```

PROPS



- Props are custom attributes that can be added to a component
- Props are used to pass data to the component
- The v-bind directive can be used to dynamically create the props

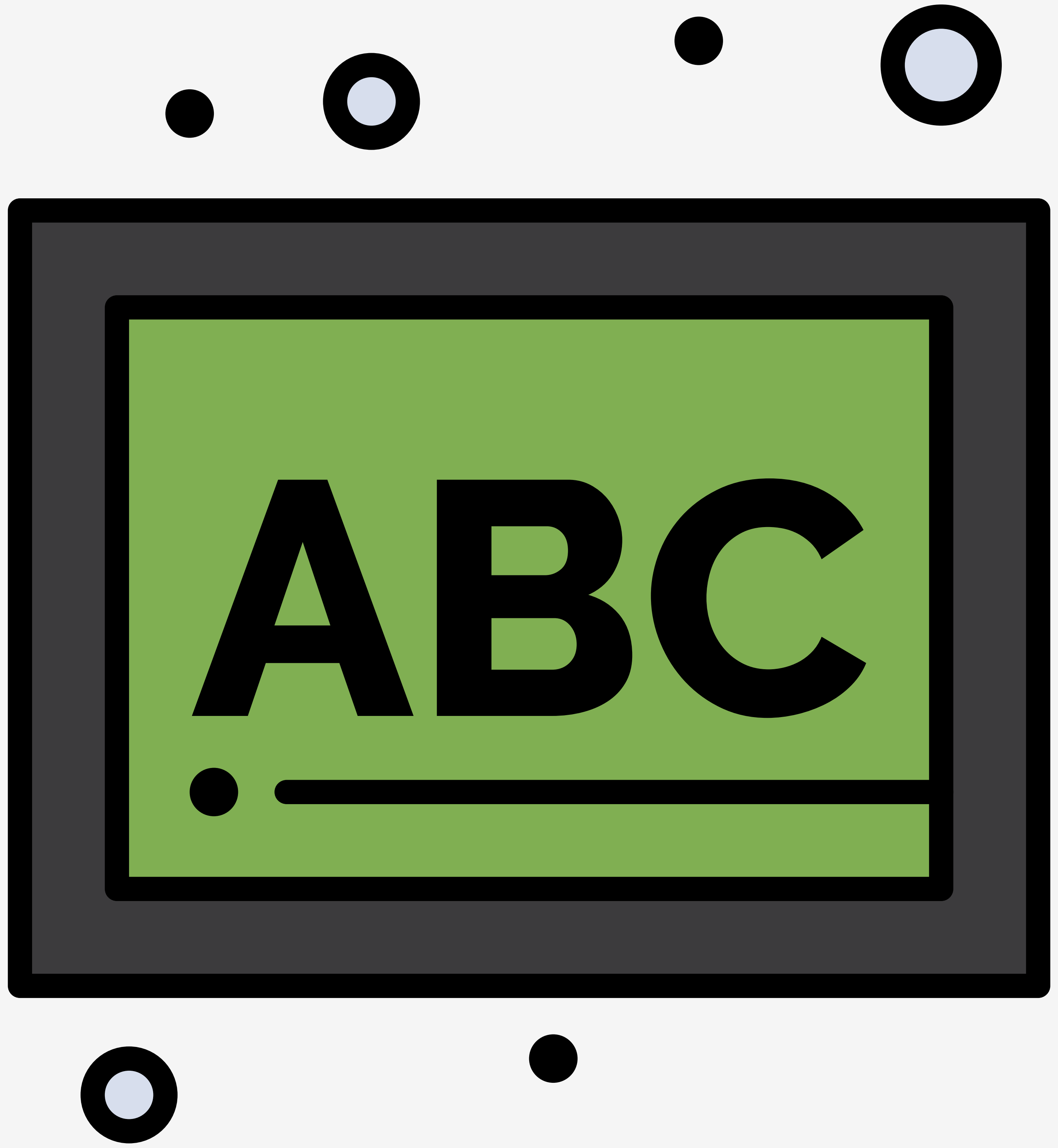
PROPS

```
<div id="app">
  <simple-button
    v-for="button in buttons"
    :text="button.text"
    :type="button.type"
  ></simple-button>
</div>
```

```
Vue.component('simple-button', {
  props: ['text', 'type'],
  template: `<button class="btn m-1" :class="'btn-'+type">
    {{ text }}
  </button>`
})

new Vue({
  el: '#app',
  data: {
    buttons: [
      {text: 'Click Me', type='primary'},
      {text: 'No, Click Me', type='success'},
      {text: 'Please, Click Me', type='danger'}
    ]
  }
})
```

SLOTS



- Slots are another to pass data to a component
- Slot data is passed between the component element tags
- Slot data can contain text, HTML or moustache syntax
- The `<slot></slot>` tag is use in the component template display the data

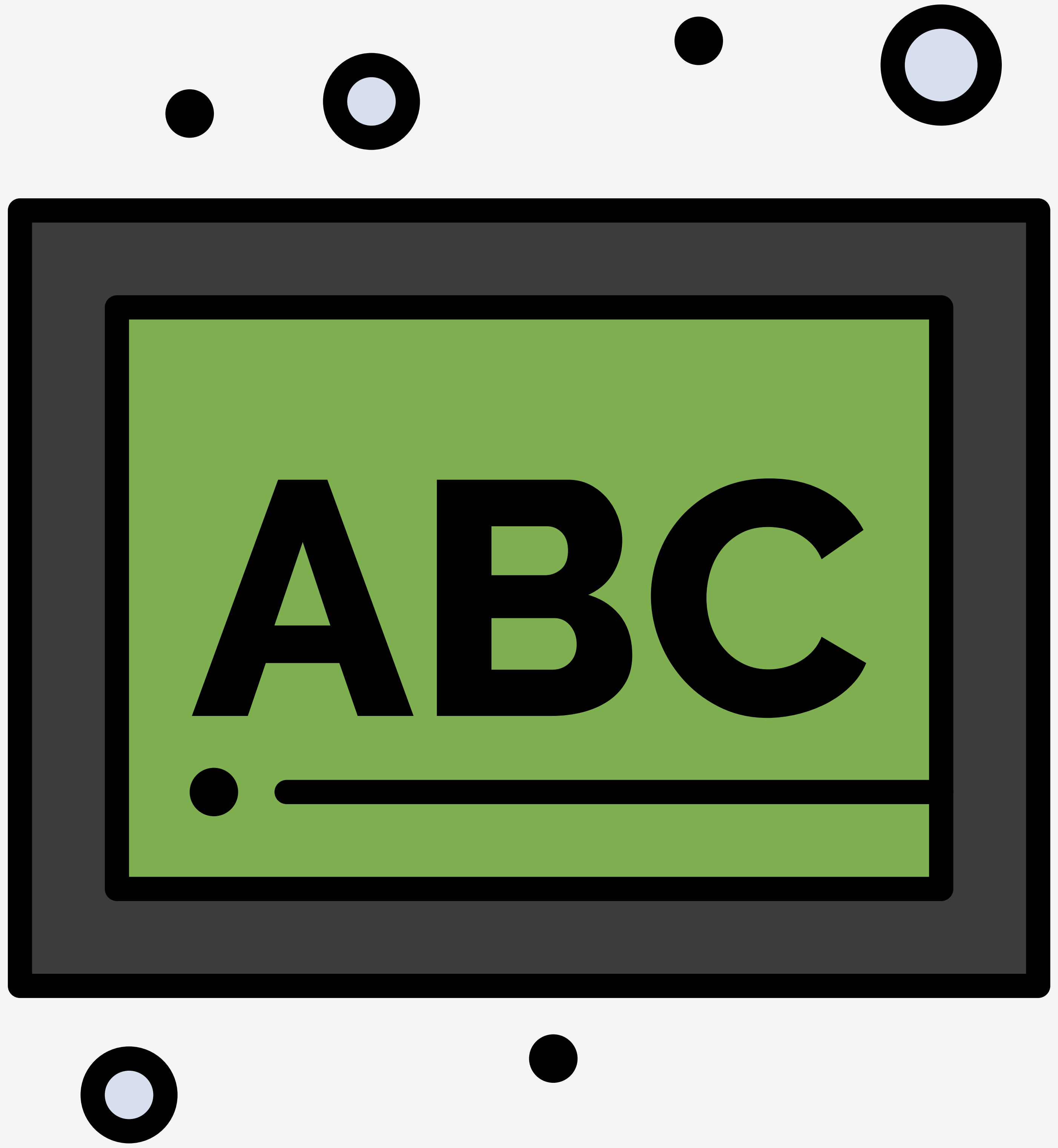
SLOTS

```
<div id="app">
  <simple-button v-for="button in buttons" :type="button.type">
    <strong>{{ button.text }}</strong>
  </simple-button>
</div>
```

```
Vue.component('simple-button', {
  props: ['type'],
  template: `
  <button class="btn m-1" :class="'btn-' + type">
    <slot></slot>
  </button>`
})

new Vue({
  el: '#app',
  data: {
    buttons: [
      { text: 'Click Me', type: 'primary' },
      { text: 'No, Click Me', type: 'success' },
      { text: 'Please, Click Me', type: 'danger' }
    ]
  }
})
```

METHODS



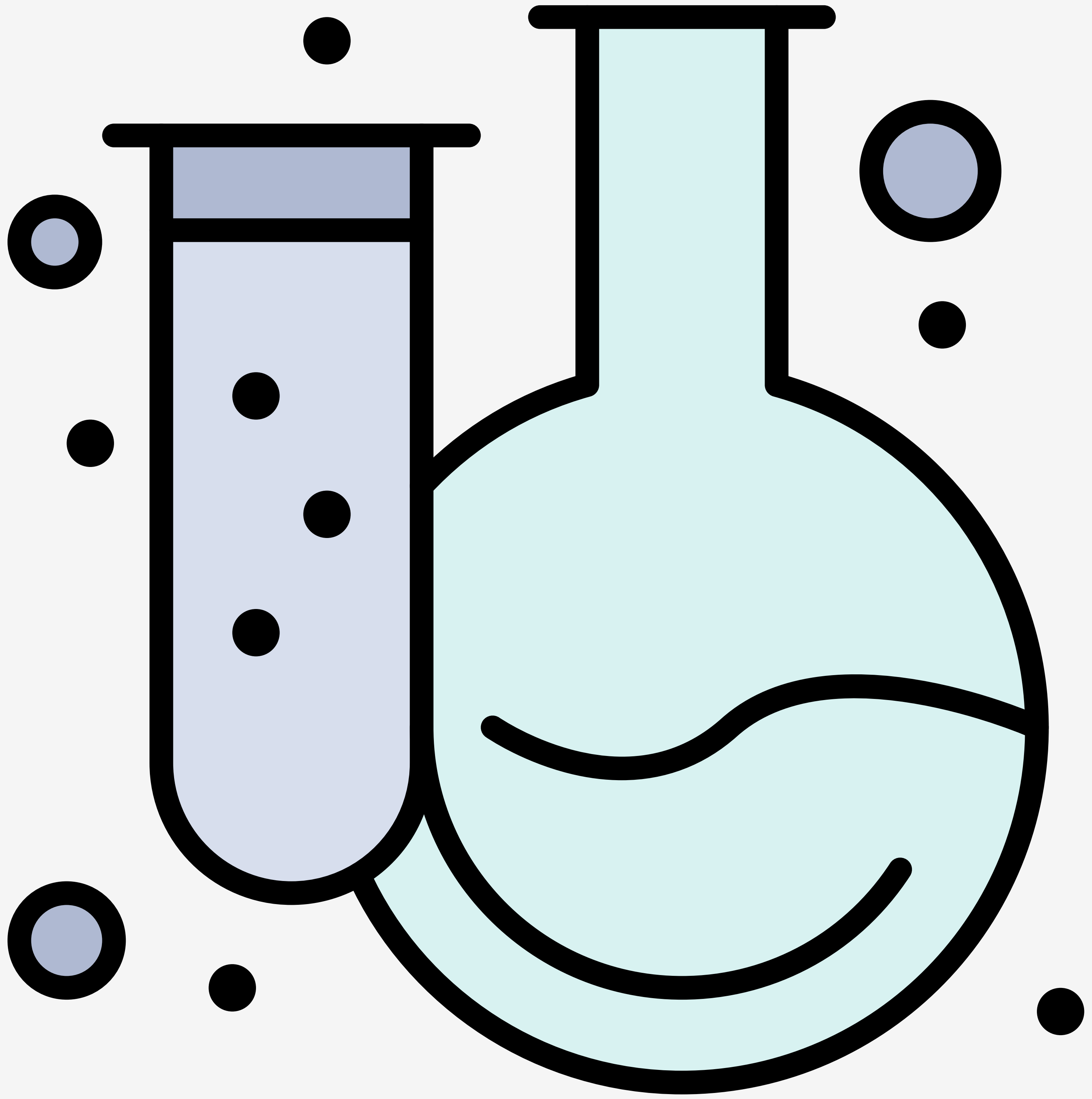
- Event handlers are added to the component template
- Components can have their own methods

METHODS

```
Vue.component('simple-button', {
  props: ['type'],
  methods: {
    action: function () {
      alert(this.type.toUpperCase())
    }
  },
  template: `
  <button class="btn m-1" :class="'btn-' + type" @click="action">
    <slot></slot>
  </button>`
})

new Vue({
  el: '#app',
  data: {
    buttons: [
      { text: 'Click Me', type: 'primary' },
      { text: 'No, Click Me', type: 'success' },
      { text: 'Please, Click Me', type: 'danger' }
    ]
  }
})
```

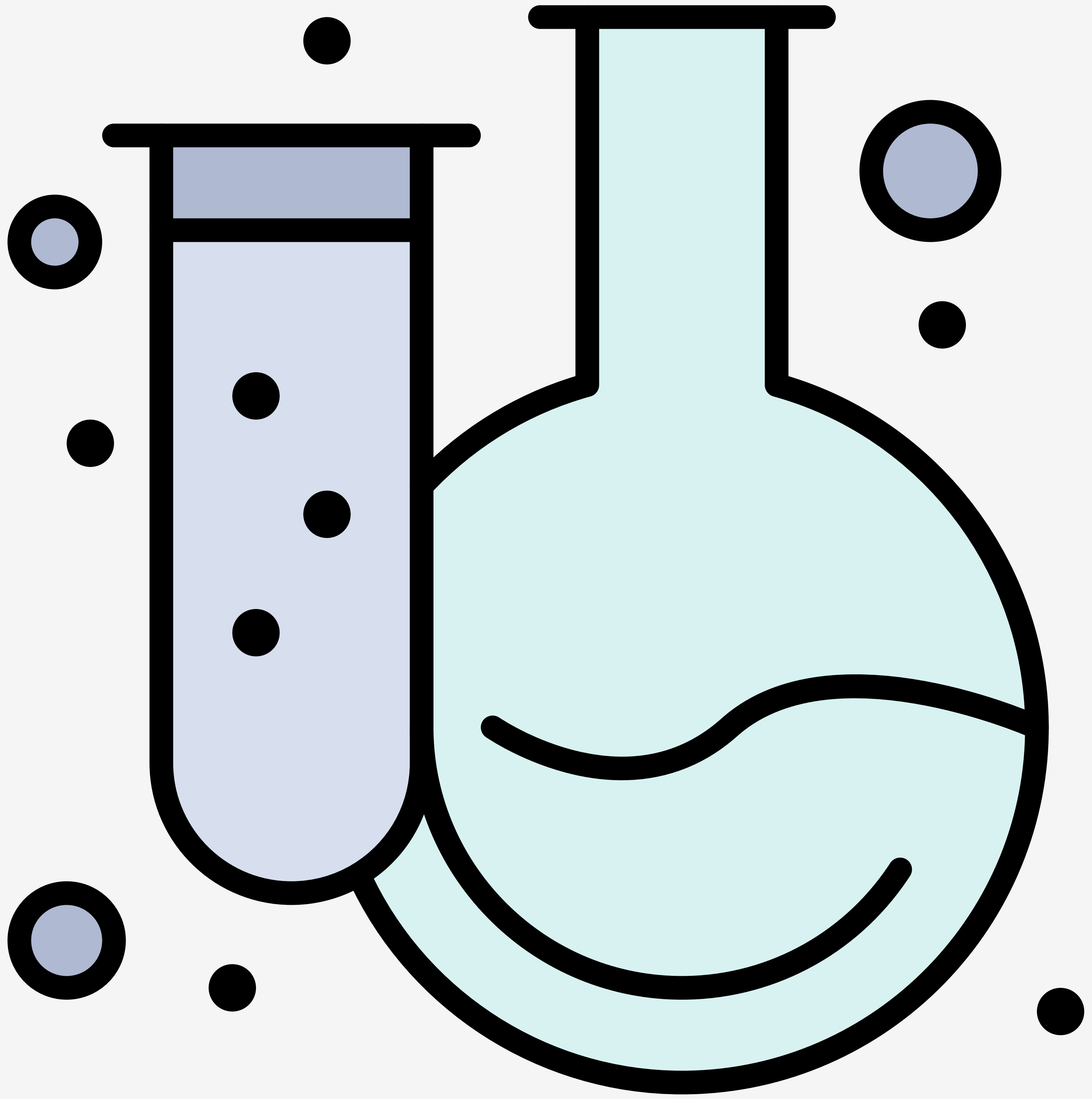
VUETIP



- *FORK THE PEN!*
- Create a reusable tooltip using a Vue component
- Submit the URL to your pen
- *DUE:* **Mon. Jan 27 @ 11:59 PM**

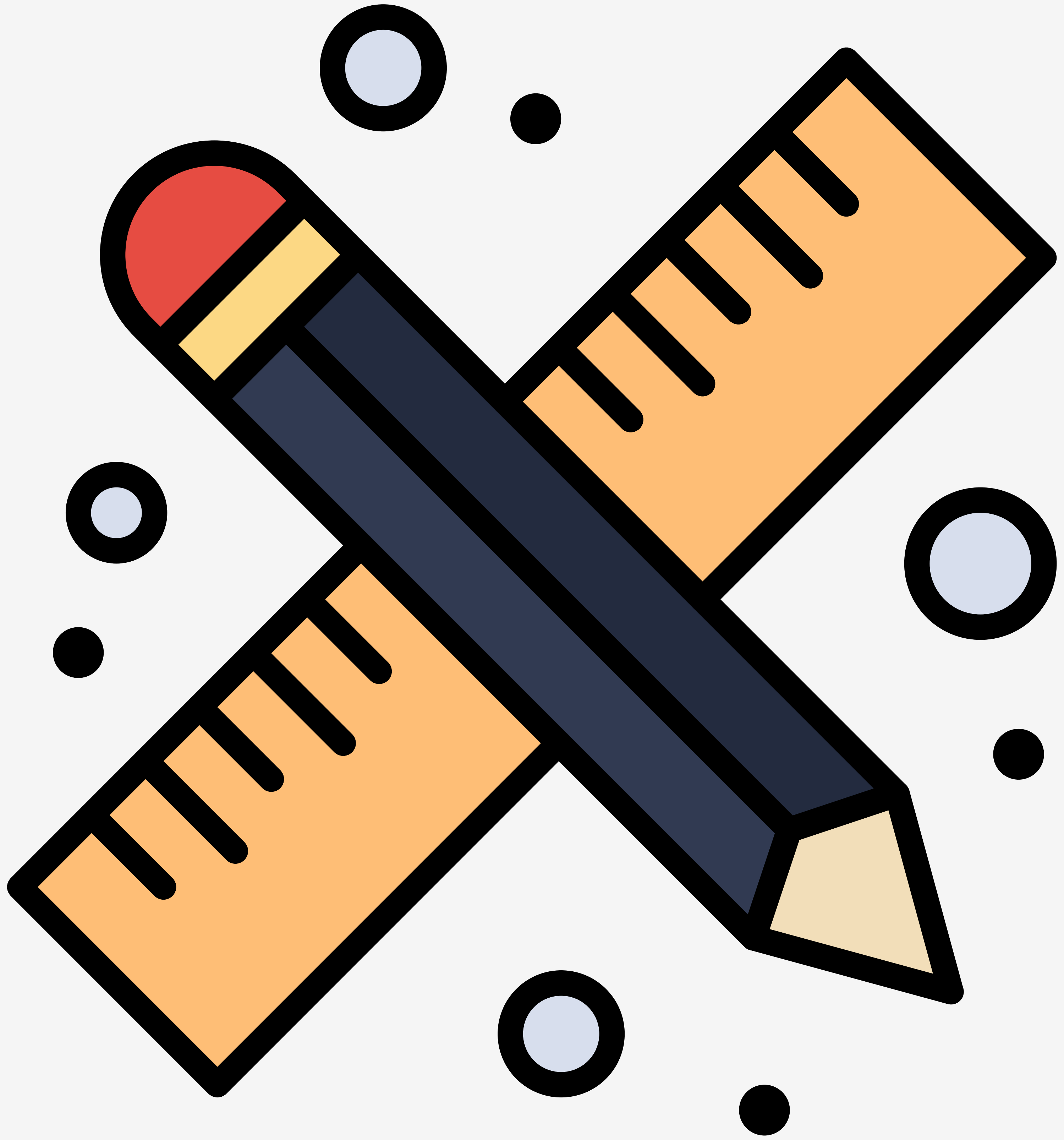
HANDS-ON

HYBRID #2



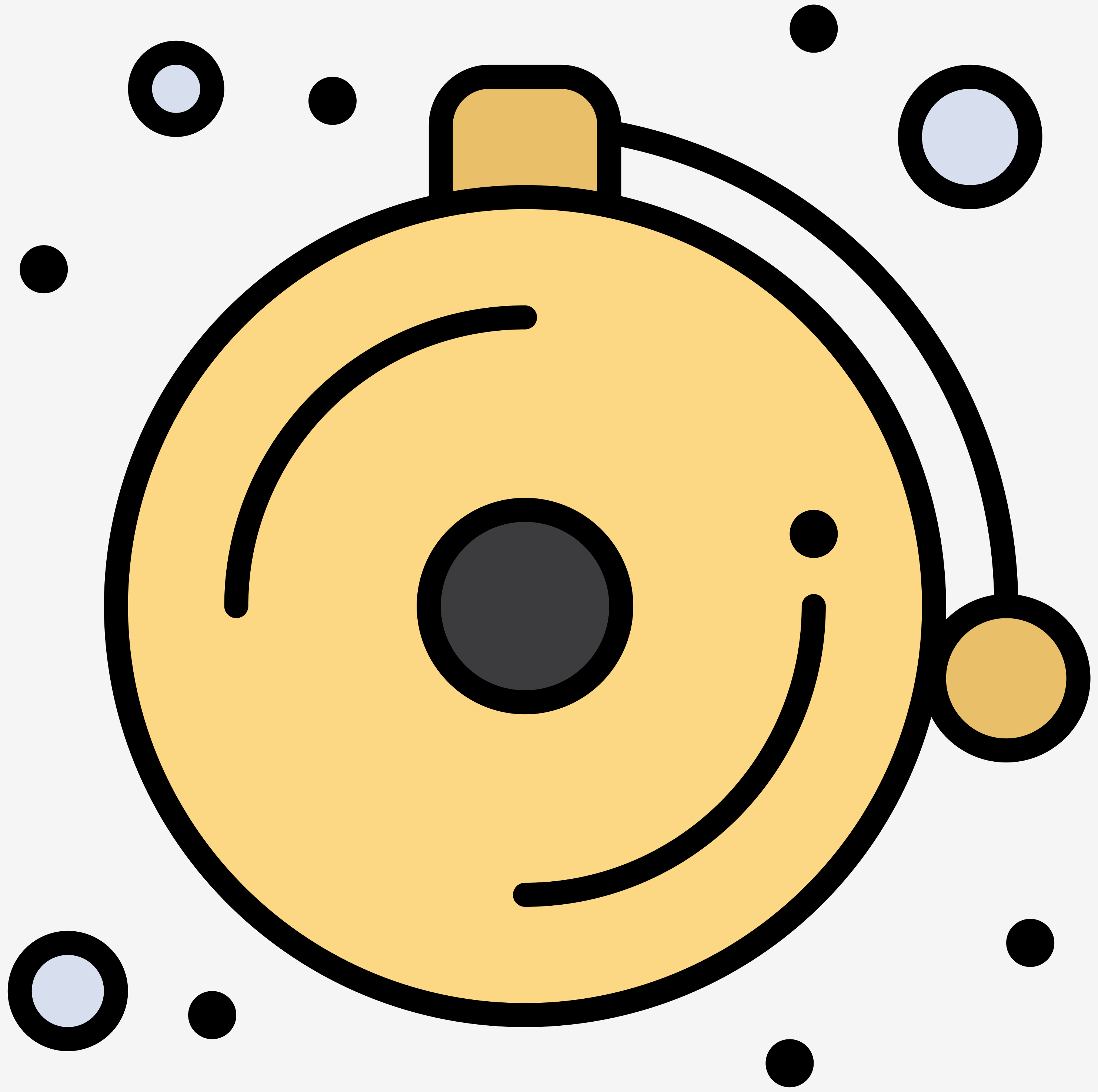
- Watch the 4th section of Learning Vue.js
- Write 2 to 3 sentences
- *DUE:* **Mon. Feb. 3 @ 11:59 PM**

GROCERY LIST



- *GITHUB CLASSROOM ASSIGNMENT*
- Create a simple grocery list web app
- Display items in their appropriate categories
- Allow user to add new items and cross out existing items
- Submit the URL to your repository
- *DUE:* **Mon. Feb 10 @ 11:59 PM**

NEXT TIME...



- Vue CLI
- Project: Event Calendar