

file-read-comp

file-read-comp is a simple Vue.js component (≥ 2.5) for reading file names/content. The component looks like a button that brings up a file dialog for selecting a file. After selection the component emits two events for the selected file name and file content.

file-read-comp can be installed via with the included `package.json` file for a local installation via the `npm install` command. **file-read-comp** depends on the [vue.js](#) framework. A demo folder is provided that used [Parcel](#) together with its associated `package.json` file to bundle together **file-read-comp** along with its [vue.js](#) dependency for a simple application. Further details are provided below for running the demo.

Props

A prop in Vue.js is a custom attribute for passing information from a parent component hosting **file-read-comp** instance(s) to a **file-read-comp** as a child component.

file-read-comp has the following prop:

`read_label` -- a label for the **file-read-comp** button (string, default: null)

`read_file` -- if the component is to read the file's content (boolean, default: false)

`css_variables` -- defines the css variables for **file-read-comp** (object, default: null)

Styling

The **css_variables** prop is a javascript object that contains any combination of css variable names as keys and associated values. The following list are the css variable names along with their default values for a quick styling of **file-read-comp** :

```
{
  file_read_comp_font_family: 'Verdana, serif',

  file_read_comp_label_font_size: '1rem',
  file_read_comp_label_font_weight: 'normal',
  file_read_comp_label_color: 'black',
  file_read_comp_label_box_shadow: '4px 4px 6px 1px rgba(0,0,0,0.4) , 2px 2px 2px 0 rgba(0,0,0,0.2)',
  file_read_comp_label_background: 'linear-gradient(to bottom, #969696 0, #969696 13%, #5f5f5f 33%, #1e1e1e 64%, #1e1e1e 100%)',
  file_read_comp_label_border_color: 'black',
  file_read_comp_label_hover_background: 'linear-gradient(to bottom, #969696 0, #1e1e1e 100%)'
}
```

Events

file-read-comp emits two events that notifies the parent component the selected file name and optionally the file's content. The event names are `file_read_comp_filename_changed` and `file_read_comp_content_changed` respectively. The parent component can listen to the this event and provide a callback for further processing. Events emitted from a child component back

to the parent is explained at [Vue Custom Events](#).

Demonstration

The demonstration of **file-read-comp** is provided in the folder named `dist` and can be viewed by hosting the `index.html` file. The demo (templated in the `App.vue` file) has one button for showing a pop-up file dialog to select a file (like the text file contained in the `dist` folder). After selection **file-read-comp** sends an event with the file name and content to the `App.vue` listener. In this demo the application is printing out the file name and content to the console.

As a suggestion, install [http-server](#) locally/globally via [npm](#) then enter the command `http-server` in the **file-read-comp** `dist` directory. From a browser enter the url: `localhost:8080/` to view the demo.

Here is some example code for using **file-read-comp** taken from `App.vue` :

```
<file-read-comp
  read_label="Read File Path"
  :read_file="read_file"
  :css_variables="css_variables"
  v-on:file_read_comp_filename_changed="filename_changed"
  v-on:file_read_comp_content_changed="content_changed">
</file-read-comp>
```

And the supporting data references:

```
data: function() {
  return {
    read_file: true,
    css_variables: {
      file_read_comp_heading_color: 'white',
      file_read_comp_label_color: 'white',
      file_read_comp_label_border_color: 'white',
      file_read_comp_label_focus_background: 'gold'
    }
  }
},
...
...
methods: {
  filename_changed: function(filename){
    console.log(`File Name: ${filename}`);
  },
  content_changed: function(content){
    console.log(`File Content: ${content}`);
  }
}
```