

Vue

- [How do I use Vue in the admin?](#)
- [How do I put Vue in dev mode?](#)
- [How do I add a custom Vue component to the admin?](#)

How do I use Vue in the admin?

When in an admin Blade view file you can use any Vue directive (such as `v-for`, `v-if`, `v-on:click` etc).

It is recommended to use the `v-on:click` syntax instead of `@click` syntax to minimise any potential conflicts between Blade & Vue.

Dev Mode

Putting Vue into dev mode allows you to use the Vue dev tools. You can put the Admin Vue into dev mode by putting this code snippet in your Blade view:

```
@push('scripts')
<script>
    window.AeroAdmin.vue.use({
        install(Vue) {
            Vue.config.devtools = true;
        },
    });
</script>
@endpush
```

Extending Vue without a Custom Component

You can extend Vue without using a custom Vue component by using `window.AeroAdmin`.

Adding Data

```
@push('scripts')
<script>
    window.AeroAdmin.addData('myVariable', '{{ old('myVariable', 'value') }}');
    window.AeroAdmin.addData('myOtherVariable', 'Example');
    window.AeroAdmin.addData('myArrayVariable', {!! json_encode(['one', 'two', 'three',
'four']) !!});
</script>
@endpush
```

Adding Computed

```
@push('scripts')
<script>
    window.AeroAdmin.addComputed('computedMethod', function () {
        return ['one', 'two', 'three', 'four'];
    });
</script>
@endpush
```

Adding Methods

```
@push('scripts')
<script>
    window.AeroAdmin.addMethod('test', function (value) {
        console.log(value);
    });

    window.AeroAdmin.addMethod('testing', function (value) {
        this.test(value);
    })
</script>
@endpush
```

Adding Watchers

```
@push('scripts')
<script>
  window.AeroAdmin.addData('foo', 'bar');

  // Normal watcher - fires on change
  window.AeroAdmin.addWatch('foo', function (value) {
    console.log(value);
  });

  // Immediate watcher - fires immediately (and on change)
  window.AeroAdmin.addWatch('foo', {
    handler: function (value) {
      console.log(value);
    },
    immediate: true
  })
</script>
@endpush
```

You can only have one watcher for a given key, if multiple are registered the last one takes priority.

Adding Listeners

```
@push('scripts')
<script>
  window.AeroAdmin.addListener('loaded', function () {
    console.log('Vue has been loaded');
  });
</script>
@endpush
```

How do I put Vue in dev mode?

You can put the Admin Vue into dev mode by putting this code snippet in your Blade view:

```
@push('scripts')
<script>
    window.AeroAdmin.vue.use({
        install(Vue) {
            Vue.config.devtools = true;
        },
    });
</script>
@endpush
```

How do I add a custom Vue component to the admin?

1. You need to add the `assetLinks` function to your modules service provider so that your modules assets are publicly linkable. An example module service provider can be found here:

<https://github.com/aerocargo/admin-example-vue/blob/master/src/ServiceProvider.php>

2. You also need to make your component `.vue` file(s) and your `main.js` file that will register all of the components. An example javascript structure can be found here:

<https://github.com/aerocargo/admin-example-vue/tree/master/resources/js>

3. You need to run `php artisan aero:link` in your terminal to link the assets.
4. Once you've got your files setup, have ran the link command and have built the javascript files (using npm: `npm run dev` or `npm run production`), you need to ensure that your module's components javascript file is loaded and the components are registered in the Blade view where you want to use the component(s).

To do this you need to include the javascript file from your module and then register the components through the `window.AeroAdmin.vue.use` method:

```
@push('scripts')
    <script src="{{ asset(mix('components.js', 'modules/aerocargo/admin-example-vue')) }}"></script>
    <script>
        window.AeroAdmin.vue.use(window.exampleAdminComponents);
    </script>
@endpush
```

After this you'll be able to use your custom Vue component(s) anywhere in the Blade file where you have registered them.