



Validations in the Composition Age



Damian Dulisz

GitHub: [@shentao](#)

Twitter: [@damiandulisz](#)

Vue.js Core Team



#140 Tuesday, May 7, 2019

Vuex v3.1.1 released. Vue-CLI v4.0-alpha is here; Managing state with Apollo and more!

Hello!

Whoa, there is so much content this week we had a really hard time picking up the articles (some of them might show up next week).

First of all, [Vuex got a new release](#) which will definitely make the NativeScript devs happy, since now it supports debugging with the remote Vue Devtools. 🎉

The [Vue-CLI v4.0.0-alpha](#) has also been released with several meaningful changes like bumping the versions on webpack-chain, core-js, ESLint, and workbox in the PWA plugin. Take it for a spin and let us know of any issues.

Also, [Natalia](#) has written an [amazing article](#) on managing state when using Apollo and is probably a must read those of you considering using Apollo in your project.

Cheers,

— *Damian Dulisz*

 PLAY THE PODCAST

The Official Vue News

We exist to provide Vue developers with the latest news and tutorials to stay up-to-date with their technology.

Get a weekly email

 Email address[SUBSCRIBE](#)

Previously known as Vue.js Newsletter

Subscribe to the podcast

[APPLE PODCASTS](#)[ANDROID](#)[STITCHER](#)[RSS](#)

Open Source Work

Vue-Multiselect

Universal select/multiselect/tagging component for Vue.js

Vuelidate

Simple, lightweight model-based validation for Vue.js

Vue-Global-Events

Register global events as a component

with Eduardo San Martin Morote @posva

FormVueLatte

Forms, Vue and Coffee ☕

with Marina Mosti @MarinaMosti

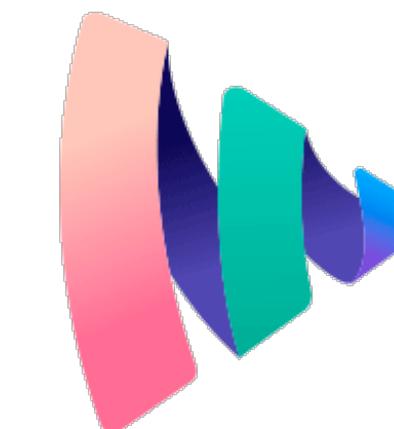


Technical Lead
@Coursedoginc

Thanks to all my supporters!

<https://github.com/users/shentao/sponsorship>

Vueconf.US



Vue School



Vue Mastery

VueJobs

And all the amazing individuals ❤️

Validations in the Composition Age

Validation libraries in Vue.js 2.x

Vee-Validate

Template Driven Validation Framework

created by Abdelrahman Awad

```
<ValidationProvider rules="min:3" v-slot="{ errors }">
  <input v-model="value" type="text">
  <span>{{ errors[0] }}</span>
</ValidationProvider>
```

Validation libraries in Vue.js 2.x

Vuelidate

Model-based Validations
created by Damian Dulisz
and Paweł Grabarz

```
export default {
  data () {
    return {
      password: '',
      repeatPassword: ''
    }
  },
  validations () {
    return {
      password: {
        required,
        minLength: minLength(7),
      },
      repeatPassword: {
        sameAs: sameAs('password'),
      }
    }
  }
}
```

Validation libraries in Vue.js 2.x

Current problems with Vuelidate:

- ❖ No lazy validation support.
- ❖ No built-in support for error messages.
- ❖ Mediocre support for asyncValidators.
- ❖ Requires validation rules for the whole data model at once.
- ❖ Hard to share the validation results between components.
Especially from a child to the parent.
- ❖ Complicated source code including several hacks to make it work with Vue 2.x, that makes it hard to maintain and contribute to.

Vue.js 3.0

Vuelidate v2.0 alpha

[vuelidate/vuelidate/tree/next](#)

```
yarn add @vuelidate/core @vuelidate/validators
```

Vuelidate v2.0

- ❖ Complete rewrite
- ❖ Built for Vue 3.0
- ❖ Half the code, quarter the complexity
- ❖ Mostly backwards compatible
- ❖ Lightweight (currently 1.6 kB min+gzip)

Vuelidate v2.0

Basic component with
password and
repeatPassword in the state.

```
export default {
  setup () {
    const password = ref('')
    const repeatPassword = ref('')

    return {
      password,
      repeatPassword,
    }
  }
}
```

Vuelidate v2.0

Let's import our new Vuelidate composition function `useVuelidate`.

It accepts and object containing the rules and an object containing the state that the rules will validate.

```
import useVuelidate from '@vuelidate/core'

export default {
  setup () {
    const password = ref('')
    const repeatPassword = ref('')
    const v$ = useVuelidate(
      rules,
      { password, repeatPassword }
    )
    return {
      password,
      repeatPassword,
      v$
    }
  }
}
```

Vuelidate v2.0

The rules, similar to Vuelidate 0.x should map 1:1 to the state that will be passed.

```
import useVuelidate from '@vuelidate/core'

export default {
  setup () {
    const password = ref('')
    const repeatPassword = ref('')
    const rules = {
      password: {
        required,
        minLength: minLength(7),
      },
      repeatPassword: {
        sameAs: sameAs(password),
      }
    }
    const v$ = useVuelidate(
      rules,
      { password, repeatPassword }
    )

    return {
      password,
      repeatPassword,
      v$
    }
  }
}
```

Vuelidate v2.0

Demo Time!

Vuelidate v2.0

Built-in Error Messages support

Vuelidate v2.0

All **21** built-in validators come with messages and params already set up.

Vuelidate v2.0

All **21** built-in validators come with messages and params already set up.

alpha

alphaNum

and

between

decimal

email

integer

ipAddress

macAddress

maxLength

maxValue

minLength

minValue

not

numeric

or

required

requiredIf

requiredUnless

sameAs

url

Vuelidate v2.0

alpha

alphaNum

and

between

decimal

email

integer

ipAddress

macAddress

maxLength

maxValue

minLength

minValue

not

numeric

or

required

requiredIf

requiredUnless

sameAs

url

You choose which one to import from `@vuelidate/validators`

Vuelidate v2.0

`@vuelidate/validators`

- ❖ Fully tree-shakeable.
- ❖ Separate package.
- ❖ Separate release cycle.
- ❖ Easy to contribute.

Vuelidate v2.0

Build your own validators!

```
import minLength from '../raw/minLength'

export default length => ({
  $validator: minLength(length),
  $message: ({ params }) => `This field should be at
    least ${params.length} long.`,
  $params: { length }
})
```

Submit a **Pull Request** to share it with others!

Forms

Forms

General Information

Section Number ?

Set Section Number

Call Number

Set Call Number

Section Type * ?

Lecture

Instruction Mode *

In Person

Campus ?

—

Part Of Term

Set Part Of Term

Meeting Patterns & Rooms

DAYS

START

END

ROOM

+ MEETING PATTERN



Forms

General Information

Section Number [Set Section Number](#)

Call Number

[Set Call Number](#)Section Type * [Lecture](#)Campus Instruction Mode * [In Person](#)

Part Of Term

[Set Part Of Term](#)

Meeting Patterns & Rooms

DAYS

START

END

ROOM

[+ MEETING PATTERN](#)

Solution:

Schema Based Forms

Common issues with schema form libraries

- ❖ Only supports predefined input components.
- ❖ Can't be used with existing UI component libraries like [Vuetify](#), [Quasar](#) or [Buefy](#).
- ❖ Limited possibilities when it comes to more complex data structures.
- ❖ Not compatible with external validation libraries like [vee-validate](#) or [vuelidate](#) that you might already use in your app.

FormVueLatte

Lightweight Schema Generated Forms

[vuelidate/formvuelatte](#)

FormVueLatte



Lightweight Schema Generated Forms

- ❖ “Bring Your Own Components” Idea
 - ❖ No built-in UI input components
 - ❖ Works with all v-model compatible components
- ❖ Super lightweight (less than 100 LoC)

Created by Marina Mosti and Damian Dulisz

FormVueLatte



Lightweight Schema Generated Forms

Demo Time!

FormVueLatte



Lightweight Schema Generated Forms

Roadmap:

- ❖ Adapters for popular UI libraries like **Vuetify**
- ❖ Parsers for popular schema formats like **JSON Schema**
- ❖ Plugin system enabling new integrations



FormVueLatte

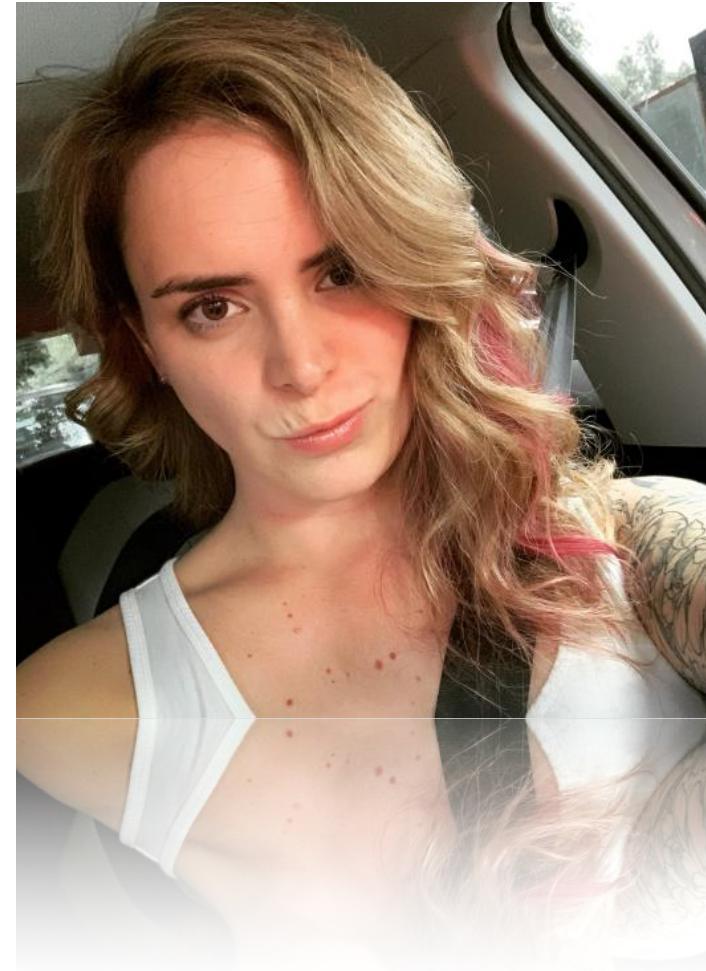


Lightweight Schema Generated Forms

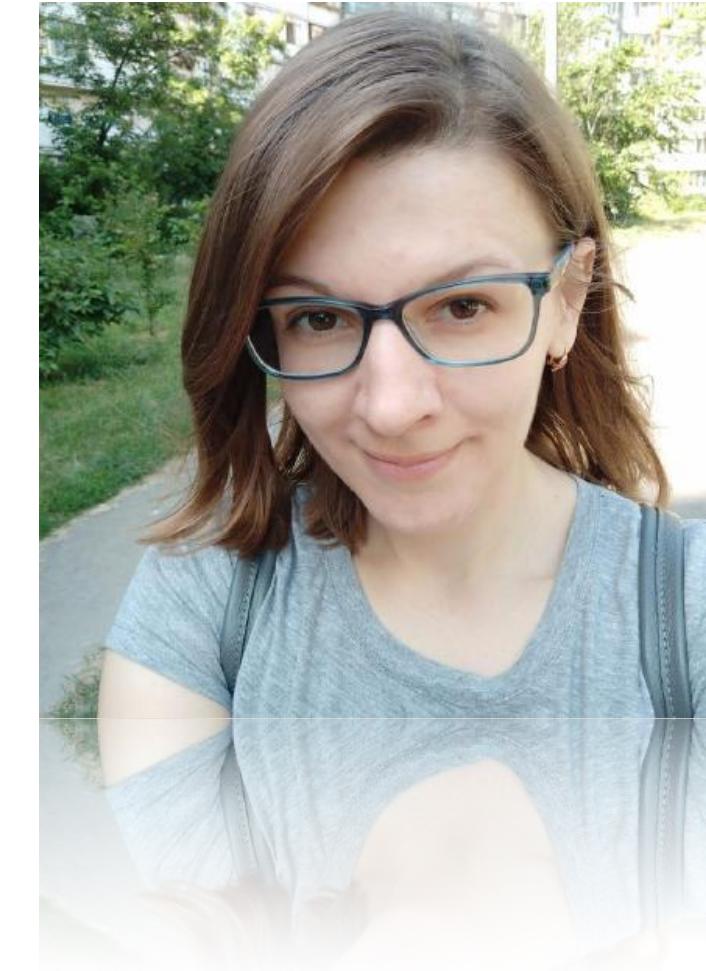
Demo Time!

Last one I promise! 😅

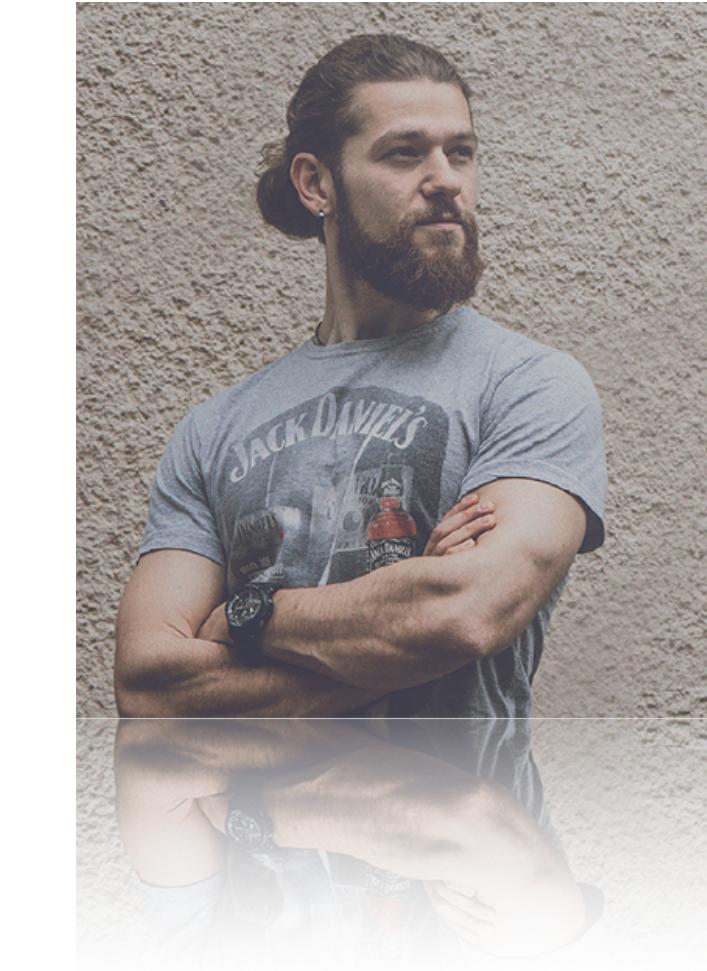
Meet the Vuelizeate team!



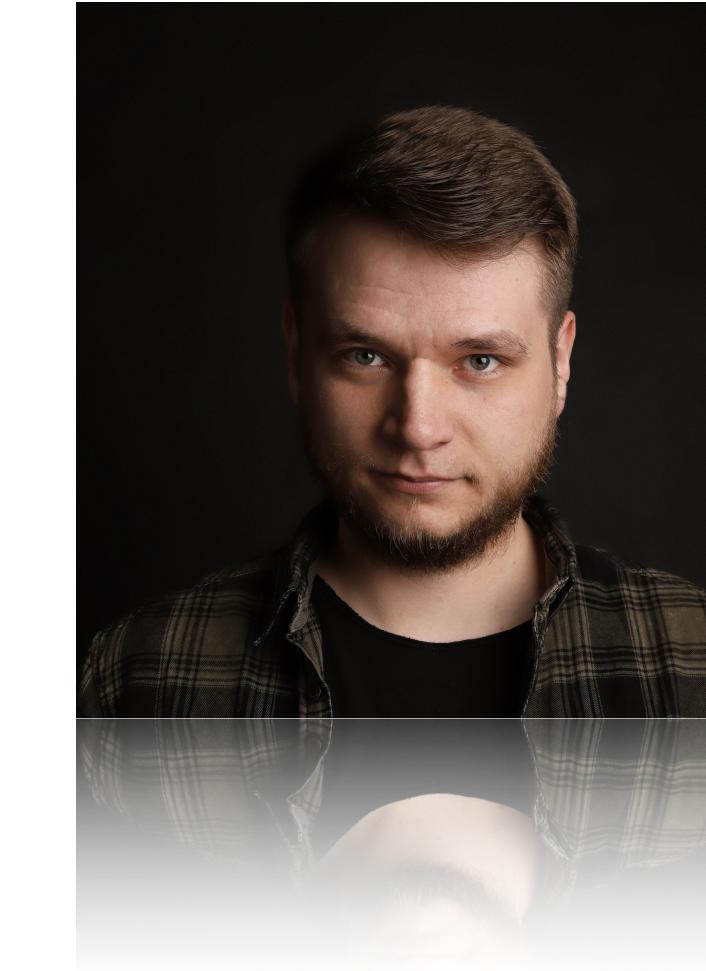
Marina Mosti
@marinamosti



Natalia Tepluhina
@n_tepluhina



Dobromir Hristov
@d_m_hristov



That guy on the
stage

Thank You!



Damian Dulisz

GitHub: @shentao

Twitter: @damiandulisz

Vue.js Core Team

Demo Code + Library Prototypes

<https://github.com/shentao/fvlidate>

<https://github.com/users/shentao/sponsorship>

Vuelidate v2.0

Nested validations

Child components with validations will report back to the ancestor component.

Meaning, the root form component will know the validation results of all child components.

