

COS30043

Interface Design and Development



Lecture 6 – Form and Validation



1

Contents

Form Validation:



- Using HTML 5 Form Attributes
- Using VueJS
- Using Vuetify



2

HTML 5 Form Validation Attributes

- required**: Specifies whether a form field needs to be filled in before the form can be submitted.
- minlength** and **maxlength**: Specifies the minimum and maximum length of textual data (strings)
- min** and **max**: Specifies the minimum and maximum values of numerical input types
- type**: Specifies whether the data needs to be a number, an email address, or some other specific preset type.
 - <input type="date" ...
 - <input type="time" ...
 - <input type="email" ...
- pattern**: Specifies a regular expression that defines a pattern the entered data needs to follow.

3

HTML 5 Form Validation Example

```
<form method="post"
action="https://mercury.swin.edu.au/it000000/formtest.php" >

<p>First Name:<input type="text" name="fName"
    required minlength="5" maxlength="8" ></p>
<p>Age:<input type="number" name="age" min="10"
    max="50" ></p>
<p>Date:<input type="date" name="date" ></p>
<p>Post Code:<input type="text" name="postcode"
    pattern="\d{4}" ></p>

<p><input type="submit" value="submit"></p>
</form>
```

4 - Interface Design and Development, © Swinburne



4

Contents

Form Validation:



- Using HTML 5 Form Attributes
- Using VueJS
- Using Vuetify

5 - Interface Design and Development, © Swinburne



5

VueJS Form Validation

- If you want VueJS completely control over the validations
 - use form's novalidate attribute to disable the HTML 5 validations supported by modern browsers
- Example: `<form novalidate >`

6 - Interface Design and Development, © Swinburne



6

VueJS Form Validation – HTML

```
<div id="app">
  <form @submit="checkForm" method="post"
    action="https://mercury.swin.edu.au/it000000/formtest.php" novalidate >
    <div v-if="errors.length">
      <p>Please correct the following error(s):</p>
      <ul> <li v-for="error in errors">{{ error }}</li> </ul>
    </div>
    <p>
      <label for="fName">First Name</label>
      <input type="text" name="fName" id="fName" v-model="fName"/>
    </p>
    <p><input type="submit" value="submit"> </p>
  </form>
</div>
```

Error message

Text input

Submit button



7 - Interface Design and Development, © Swinburne

7

VueJS Form Validation – Vue JS

```
const app = Vue.createApp({
  data() { return { fName: '', errors: [] } },

  methods: {
    checkForm: function(e) {
      this.errors = []; var result = true;
      if ( !this.fName ) {
        this.errors.push("First name required")
        result = false;
      }
      if (!result)
        e.preventDefault(); //prevent form submission
    }
  }
})
app.mount('#app')
```

8

Contents

Form Validation:

- Using HTML 5 Form Attributes
- Using VueJS
- Using Vuetify



9 - Interface Design and Development, © Swinburne



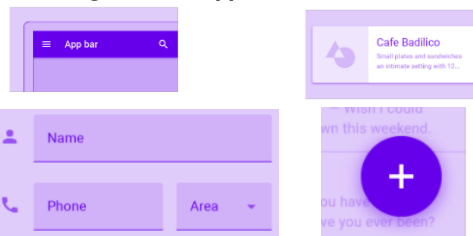
9

Vuetify

- Complete UI framework built on top of Vue.js
- Vuetify is a Vue UI Library with Material design components
- **Material design** (<https://material.io/>) was created by Google to help teams build high-quality digital experiences for Android, iOS, Flutter, and the web.
- Material Design is inspired by the physical world and its textures, including how they reflect light and cast shadows. Material surfaces reimagine the mediums of paper and ink.
- There are different components for creating user interface, such as cards, floating buttons, app bar, text fields...



10 - Interface Design and Development, ©



10

Vuetify

- <https://vuetifyjs.com/>




Vue Component Framework

Vuetify is a no design skills required UI Library with beautifully handcrafted Vue Components.

 GET STARTED

 WHY VUETIFY?

 GITHUB



11 - Interface Design and Development, © Swinburne

11

Vuetify - Installation

- To use Vuetify, in the <head> section, use <link> link to
 - Material design icon
 - Vuetify css
- Use <script> link to
 - vue.js
 - vuetify.js

12

Template for VueJS with Vuetify

```
<!DOCTYPE html>
<html>
<head>
<!-- link to material design -->
<link href="https://cdn.jsdelivr.net/npm/@mdi/font@6.x/css/materialdesignicons.min.css"
rel="stylesheet">
<!-- link to vuetify css -->
<link href="https://cdn.jsdelivr.net/npm/vuetify@3.1.11/dist/vuetify.min.css"
rel="stylesheet">

<meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>
  <div id="app">
    <v-app>
      <v-main>
        <v-container>Hello world</v-container>
      </v-main>
    </v-app>
  </div>
```

The v-app component is a mandatory wrapper required for your application to work properly.

13

Template for VueJS with Vuetify (continued)

```
<!-- link to vue js -->
<script src="https://unpkg.com/vue@3/dist/vue.global.js"></script>
<!-- link to vuetify js -->
<script src="https://cdn.jsdelivr.net/npm/vuetify@3.1.11/dist/vuetify.min.js">
</script>


<script>
  const { createApp } = Vue
  const { createVuetify } = Vuetify
  const app = createApp()
  const vuetify = createVuetify()
  app.use(vuetify).mount('#app')
</script>
</body>
</html>
```

14

Vuetify Example

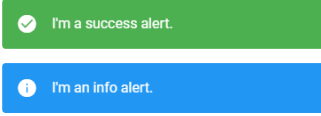
```
<v-container fluid>
  <v-btn color="primary" >
    <v-icon>mdi-pencil</v-icon> <!-- mdi: material design icon -->
  </v-btn>

  <v-btn color="error" >
    <v-icon>mdi-alarm</v-icon>
  </v-btn>
</v-container>
```



```
<v-container fluid>
  <v-alert type="success">
    I'm a success alert.
  </v-alert>

  <v-alert type="info">
    I'm an info alert.
  </v-alert>
</v-container>
```



15 - Interface Design and Development, © Swinburne



15

Vue JS Form Validation

- We can validate forms using custom Vue logic
- We can also use Vuetify to build custom rules on each component to validate forms
- We can also use Vue frameworks like Vee-validate and vuelidate for form validation

16 - Interface Design and Development, © Swinburne



16

Using Vuetify for form validation

- Vuetify has a component **v-form** which makes it easy to add validation to form inputs.
- All the input components has a *rules* prop which accepts a mixed array of types *function*, *boolean* and *string* .



17 - Interface Design and Development, © Swinburne

17

Vuetify Form Validation Example (HTML)

```
<v-form ref="myForm" method="post"
action="http://mercury.swin.edu.au/it000000/formtest.php">

  <v-text-field name="firstName" v-model="firstName"
    :rules="nameRules" label="First Name"    >
  </v-text-field>

  <v-text-field name="email" v-model="email" type="email"
    :rules="emailRules" label="Email"    >
  </v-text-field>

  <v-btn type="submit" color="success">Submit</v-btn>

</v-form>
```

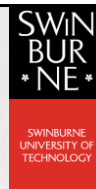
18

Vuetify From Validation Example (JS)

```
const { createApp } = Vue
const { createVuetify } = Vuetify
const vuetify = createVuetify( )
const app = createApp({
  data: () => ({
    firstName: '', email: '',
    nameRules: [
      v => !!v || 'Name required',
      v => (v && v.length <= 10) || 'Name must be less than 10
characters'
    ],
    emailRules: [
      v => !!v || 'E-mail is required',
      v => /[a-z0-9._%+-]+@[a-z0-9.-]+\.[a-z]{2,3}$/.test(v) || 'E-
mail must be valid',
    ]
  })
})
app.use(vuetify).mount('#app')
```

19

WHAT'S NEXT?
- API 1



20