



oneM2M Web Application

JiEun Lee (je.lee@keti.re.kr)

Korea Electronics Technology Institute

2020.11.13

The project "International Digital Cooperation - ICT Standardisation" is funded by the European Union



Motivation

- When a developer implements a web-app, it can be run on multiple operating systems and devices (e.g. Android, iOS, Windows, Mac)
- This tutorial provides simple web-based application that implements oneM2M APIs that we've learned so far, so developers later can extend APIs, GUIs and services logics to have their own web-based applications

Outline



- Prerequisites
- Introduction to Vue.js
- Environment Setup
- Vue.js 101
- Tasting oneM2M Web App

Prerequisites

- Node.js
 - <https://nodejs.org>
- Vue.js
 - <https://vuejs.org>
- Visual Studio Code
 - <https://code.visualstudio.com>



Introduction to Vue.js

Vue.js is..

- Web Frontend Framework
- A framework that allows user to build a component-based SPA
 - Component: Various UI elements structured to be reusable
 - SPA(Single page application): Loading only the required area within a single page

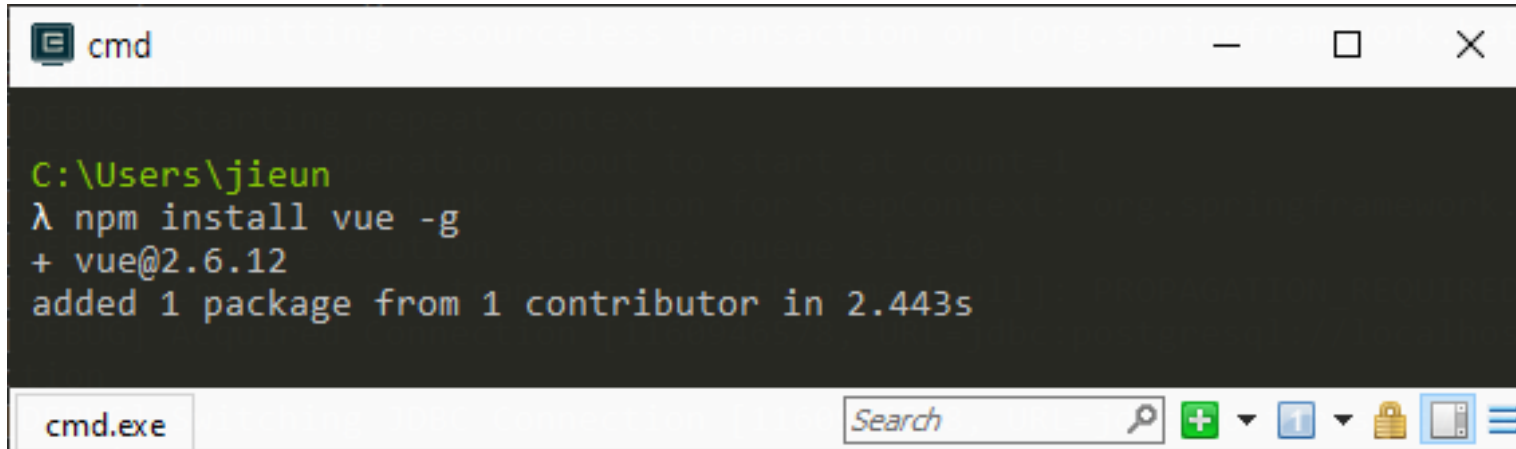




Environment Setup

Vue.js Installation

- Install Vue from CMD
 - npm install vue -g



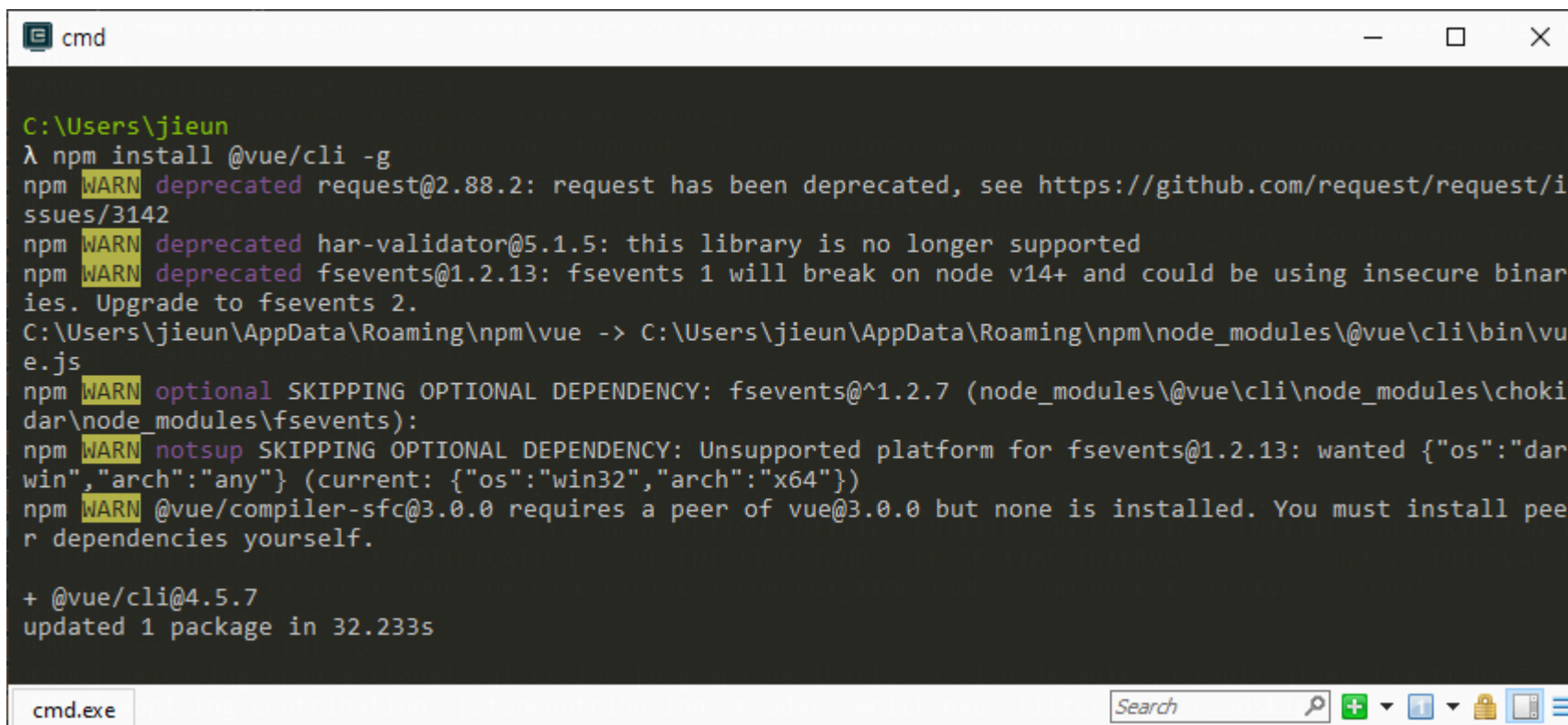
```
cmd

C:\Users\jieun
λ npm install vue -g
+ vue@2.6.12
added 1 package from 1 contributor in 2.443s
```

The screenshot shows a Windows Command Prompt window titled 'cmd'. The prompt is at 'C:\Users\jieun'. The user has entered the command 'λ npm install vue -g'. The output shows that 'vue@2.6.12' was successfully installed, adding 1 package from 1 contributor in 2.443 seconds. The taskbar at the bottom shows 'cmd.exe' and a search bar.

Vue.js Installation

- Install vue-cli from CMD
 - npm install @vue/cli -g



```
cmd
C:\Users\jieun
λ npm install @vue/cli -g
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated fsevents@1.2.13: fsevents 1 will break on node v14+ and could be using insecure binaries. Upgrade to fsevents 2.
C:\Users\jieun\AppData\Roaming\npm\vue -> C:\Users\jieun\AppData\Roaming\npm\node_modules\@vue\cli\bin\vue.js
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@^1.2.7 (node_modules\@vue\cli\node_modules\chokidar\node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
npm WARN @vue/compiler-sfc@3.0.0 requires a peer of vue@3.0.0 but none is installed. You must install peer dependencies yourself.

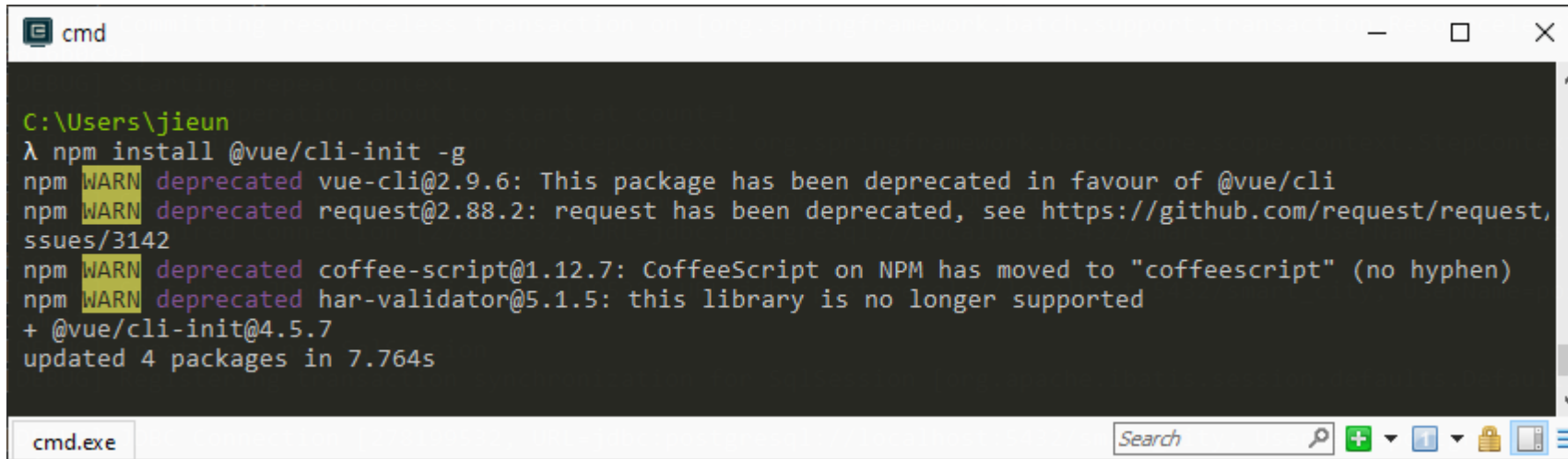
+ @vue/cli@4.5.7
updated 1 package in 32.233s
```



Vue.js 101

Vue Project Creation

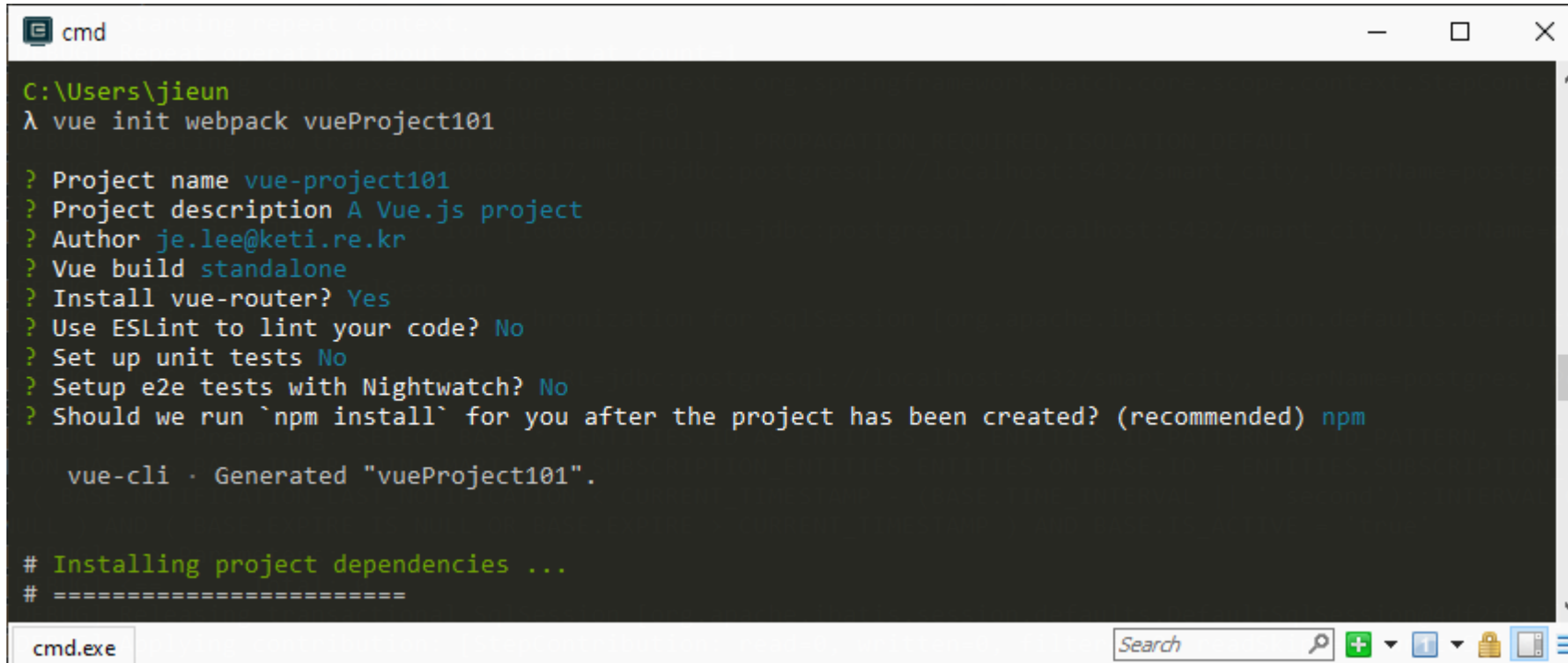
- Install vue-cli-init from CMD
 - `npm install @vue/cli-init -g`



```
cmd
C:\Users\jieun
λ npm install @vue/cli-init -g
npm WARN deprecated vue-cli@2.9.6: This package has been deprecated in favour of @vue/cli
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request,
ssues/3142
npm WARN deprecated coffee-script@1.12.7: CoffeeScript on NPM has moved to "coffeescript" (no hyphen)
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
+ @vue/cli-init@4.5.7
updated 4 packages in 7.764s
```

Vue Project Creation

- Create vue project('vueProject101')
 - vue init webpack 'projectName'



```
cmd
C:\Users\jieun
λ vue init webpack vueProject101

? Project name vue-project101
? Project description A Vue.js project
? Author je.lee@keti.re.kr
? Vue build standalone
? Install vue-router? Yes
? Use ESLint to lint your code? No
? Set up unit tests No
? Setup e2e tests with Nightwatch? No
? Should we run `npm install` for you after the project has been created? (recommended) npm

vue-cli · Generated "vueProject101".

# Installing project dependencies ...
# =====
```

Running the Project

- Go to Project folder and run
 - `cd 'folderName'`
 - `npm run dev`
- <http://localhost:8080/#/>

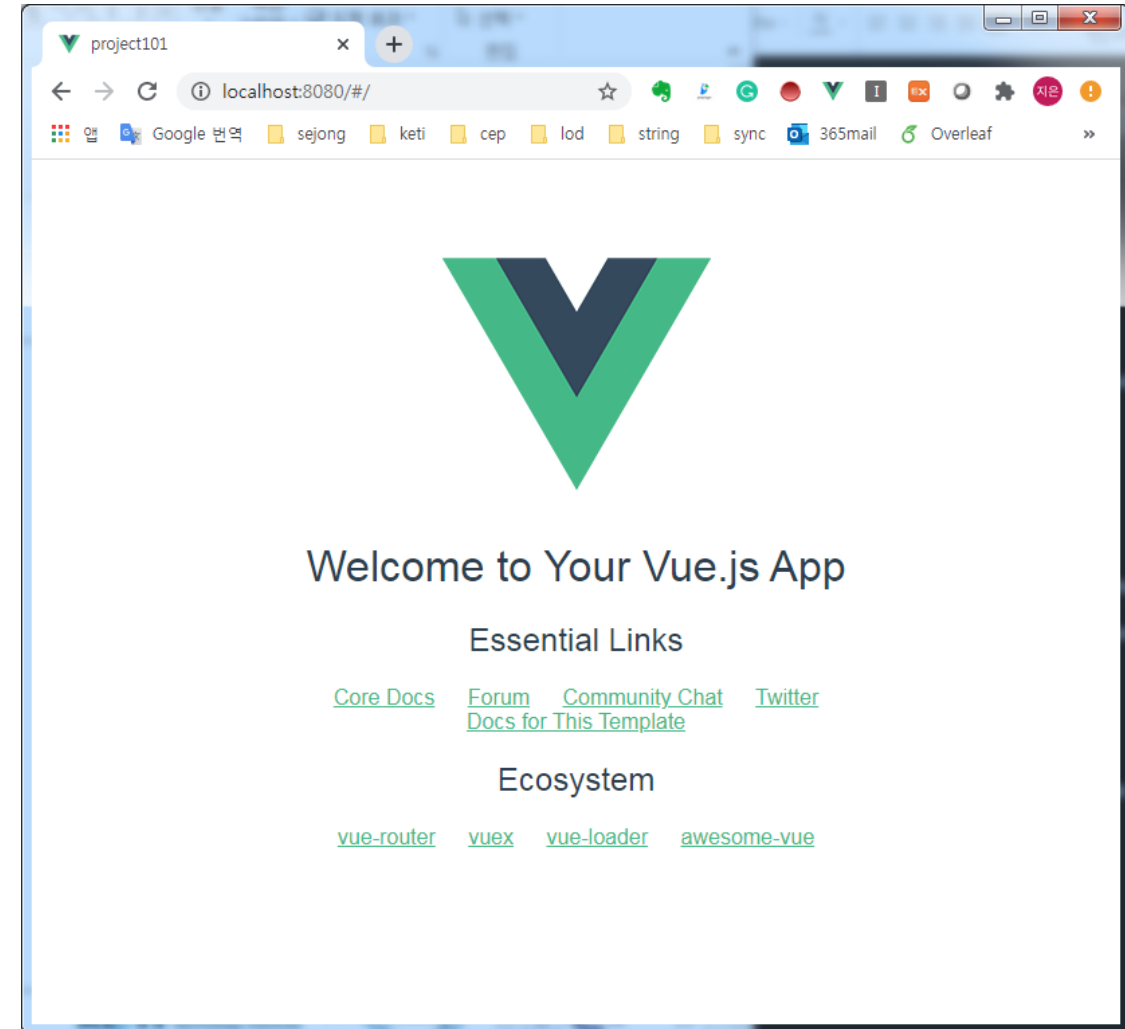
```
npm
C:\Users\jieun
λ cd vueProject101\

C:\Users\jieun\vueProject101
λ npm run dev

> vue-project101@1.0.0 dev C:\Users\jieun\vueProject101
> webpack-dev-server --inline --progress --config build/webpack.dev.conf.js

12% building modules 21/27 modules 6 active ...Users\jieun\vueProject101\src\App.vue{
parser: "babylon" } is deprecated; we now treat it as { parser: "babel" }.
95% emitting

DONE Compiled successfully in 2654ms
16:17:06
Your application is running here: http://localhost:8080
```

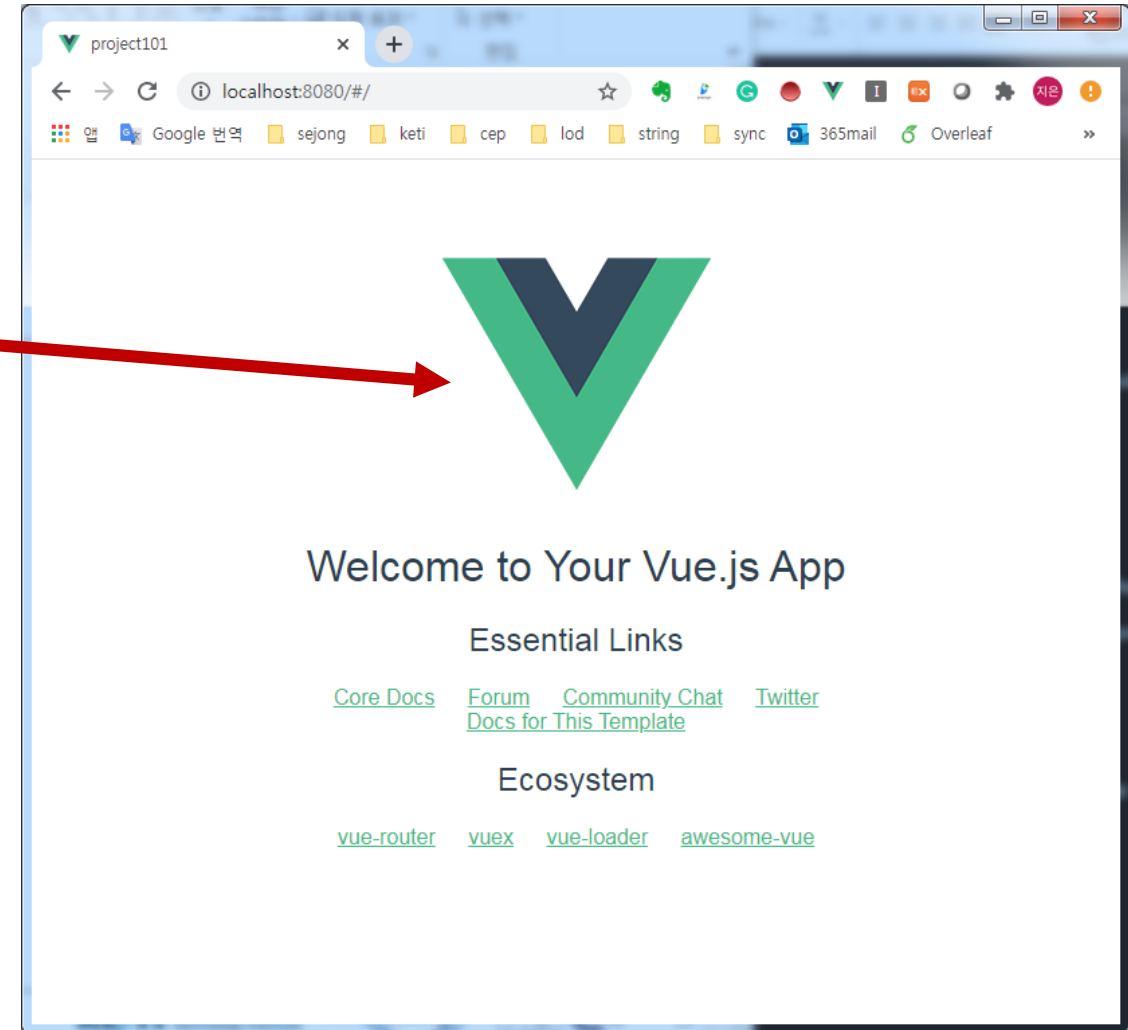


Vue Project 101

- src/App.vue

```
<template>
  <div id="app">
    
    <router-view/>
  </div>
</template>
```

- src/router/index.js



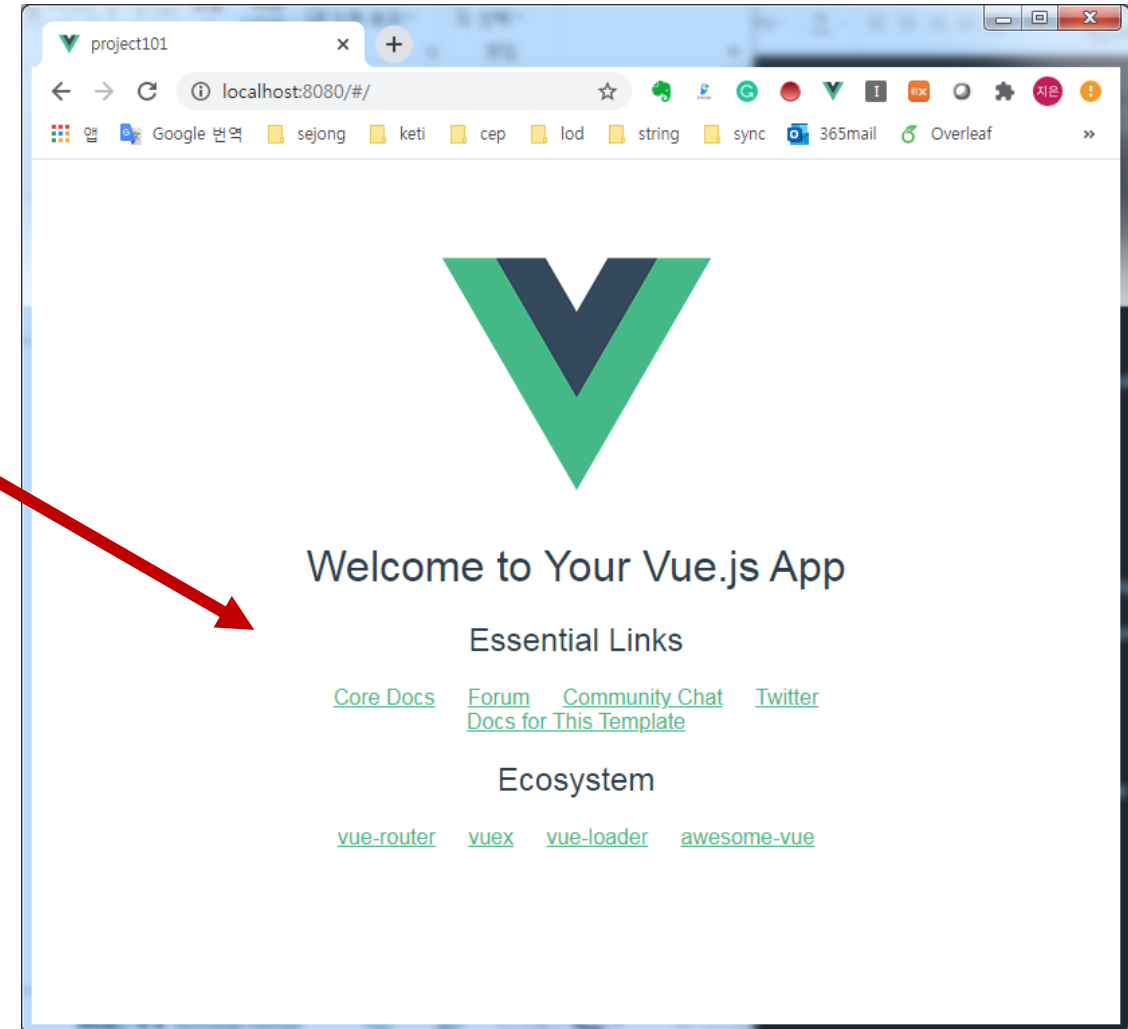
Vue Project 101

- src/router/index.js

```
import Vue from 'vue'
import Router from 'vue-router'
import HelloWorld from '@components/HelloWorld'

Vue.use(Router)

export default new Router({
  routes: [
    {
      path: '/',
      name: 'HelloWorld',
      component: HelloWorld
    }
  ]
})
```



Vue Project 101

- src/components/ButtonEvent.vue

```
<template>
  <div>
    <button>click me</button>
  </div>
</template>

<script>
export default {
}
</script>

<style scoped>

</style>
```

- src/router/index.js

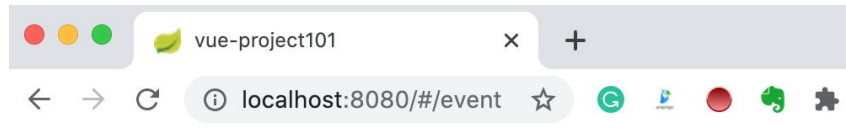
```
import HelloWorld from '../components/HelloWorld'
import ButtonEvent from '../components/ButtonEvent'

Vue.use(Router)

export default new Router({
  routes: [
    {
      path: '/',
      name: 'HelloWorld',
      component: HelloWorld
    },
    {
      path: '/event',
      name: 'ButtonEvent',
      component: ButtonEvent
    }
  ]
})
```

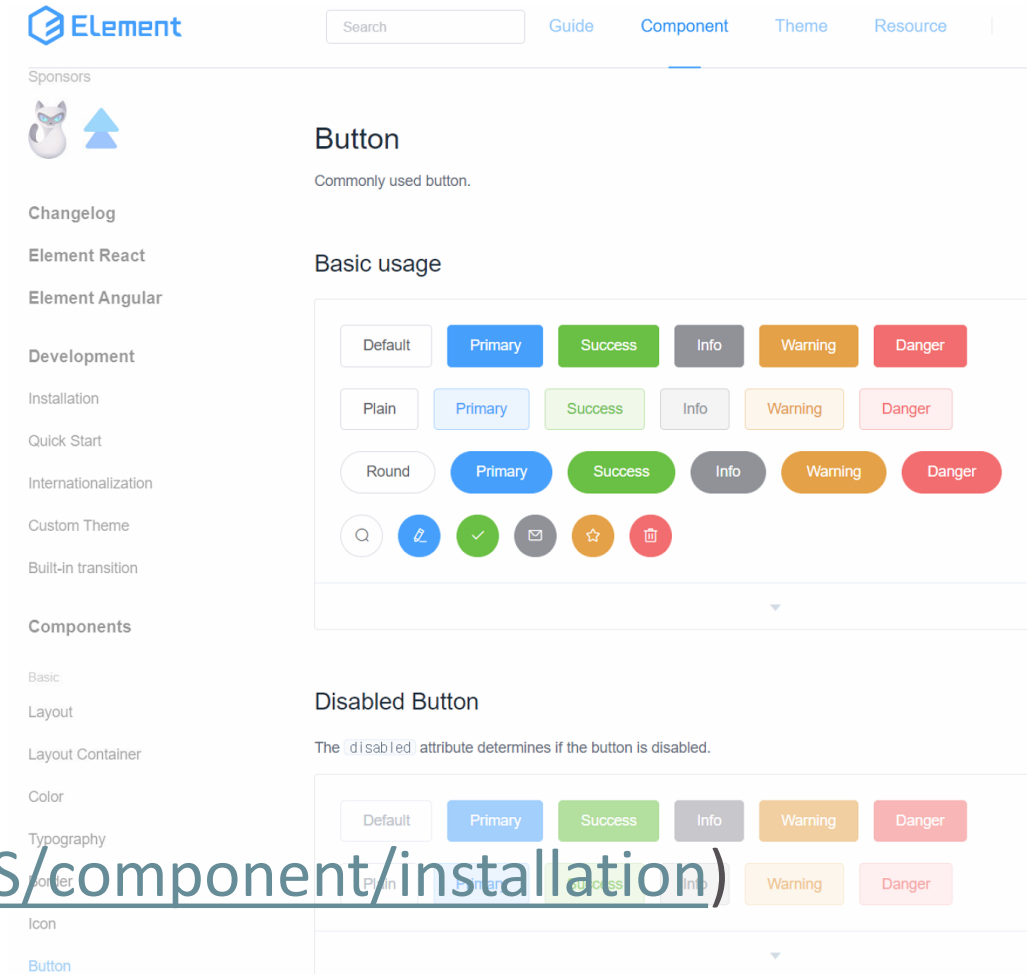

Vue Project 101

- <http://localhost:8080/#/event>



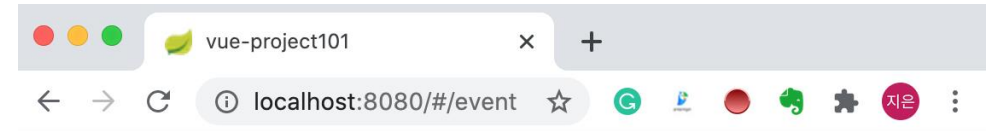
click me

- Element UI (<https://element.eleme.io/#/en-US/component/installation>)



Vue Project 101

- `npm i element-ui`
- `src/main.js`
 - `import ElementUI from 'element-ui';`
 - `import 'element-ui/lib/theme-chalk/index.css';`
 - `Vue.use(ElementUI);`
- `src/component/ButtonEvent.vue`
 - `<el-button type="success">click me</el-button>`

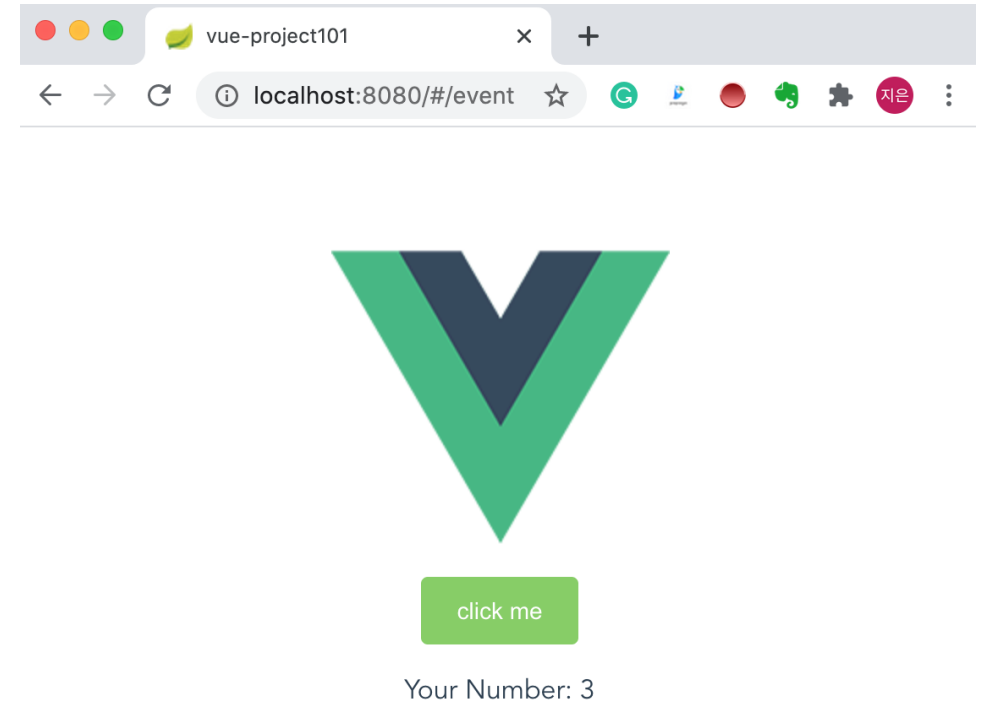


Event Handling

- src/component/ButtonEvent.vue

```
<template>
  <div>
    <el-button v-on:click="counter += 1" type="success">
      click me</el-button>
    <p>Your Number: {{counter}}</p>
  </div>
</template>

<script>
export default {
  data() {
    return {
      counter: 0
    }
  }
}
```

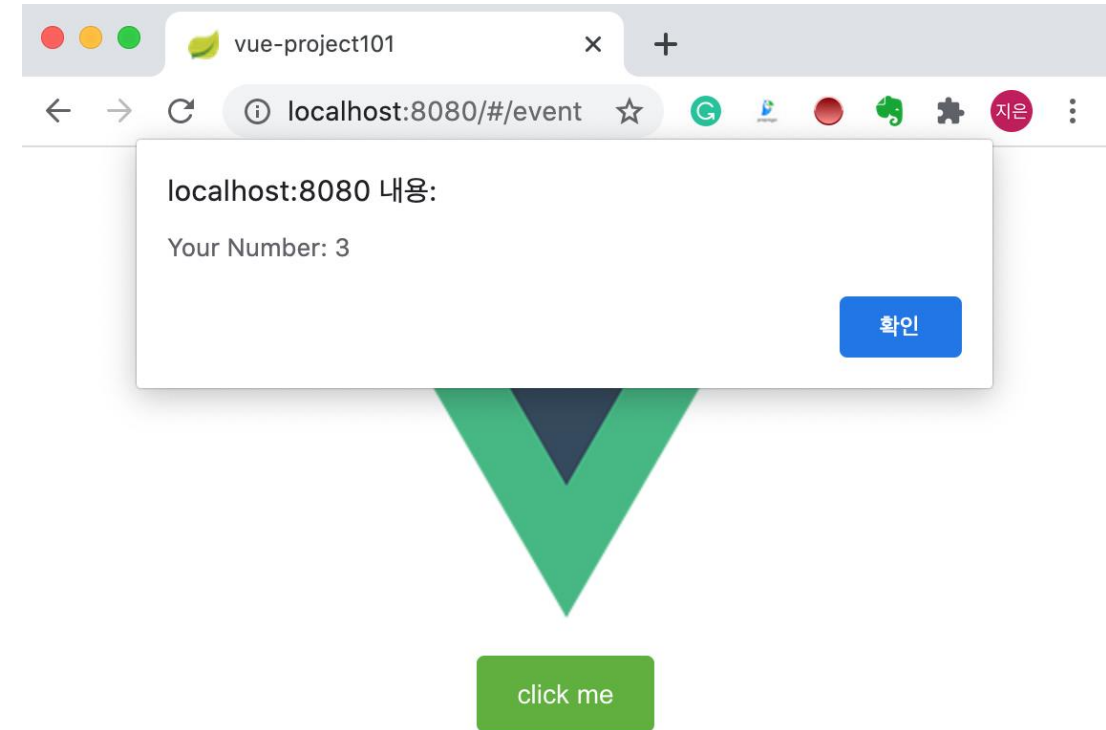


Event Handling

- src/component/ButtonEvent.vue

```
<template>
  <div>
    <el-button v-on:click="click" type="success">
      click me</el-button>
    </div>
  </template>

<script>
export default {
  data() {
    return {
      counter: 0
    }
  },
  methods: {
    click: function() {
      this.counter += 1
      alert('Your Number: ' + this.counter)
    }
  }
}
```





Tasting oneM2M Web App

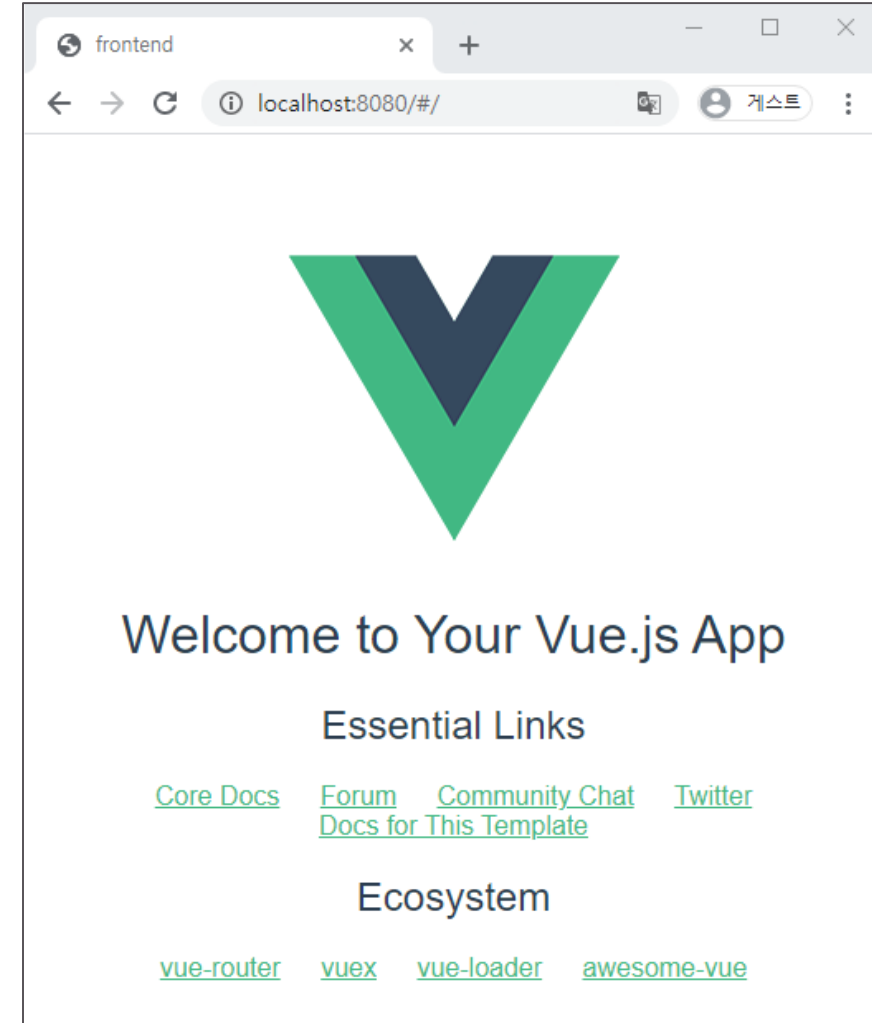
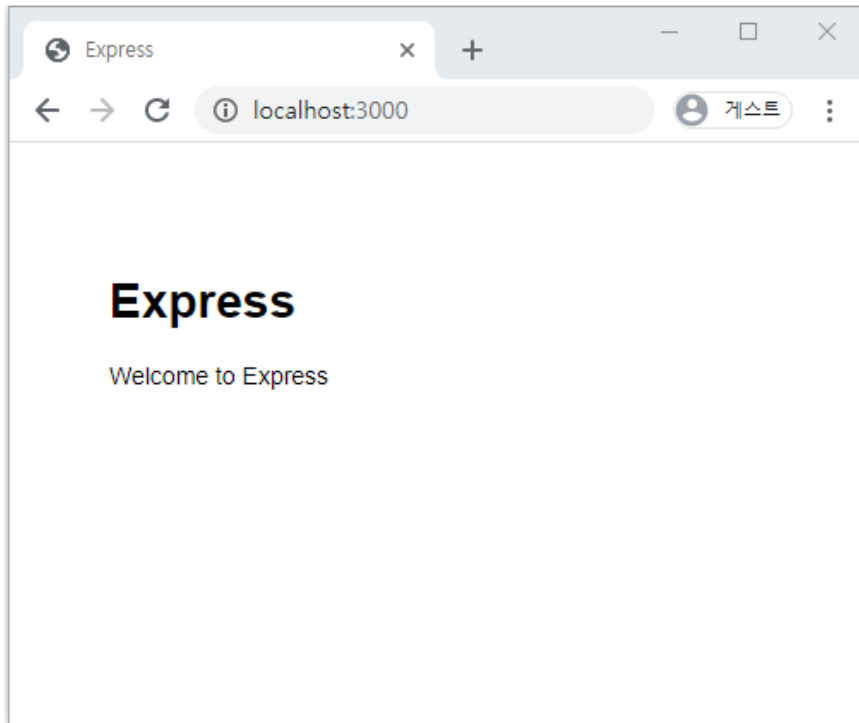
oneM2M Vue Project



- Create project folder
 - mkdir oneM2MApplication
 - cd oneM2MApplication
- Build frontend
 - git clone <https://github.com/IoTKETI/oneM2MApplicationFrontend>
 - cd oneM2MApplicationFrontend
 - npm install
 - npm run dev
- Build backend
 - git clone <https://github.com/IoTKETI/oneM2MApplicationBackend>
 - cd oneM2MApplicationBackend
 - npm start

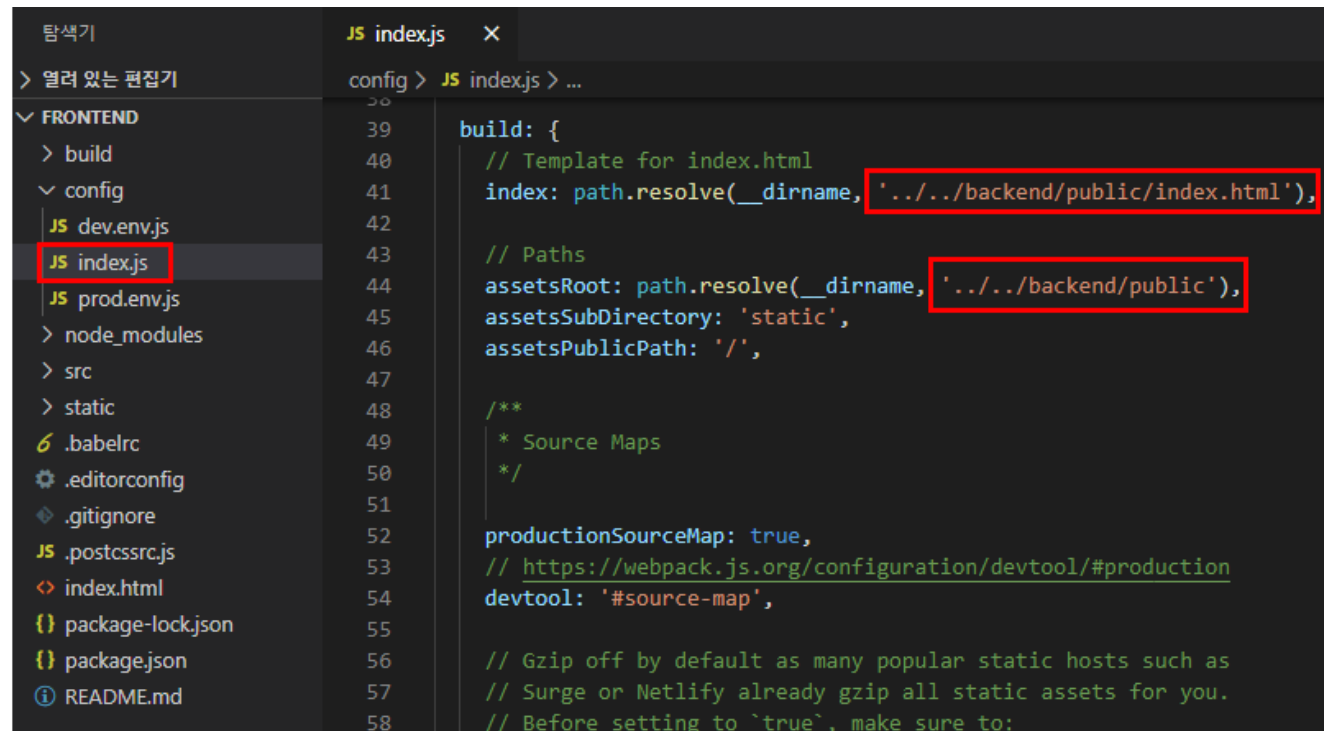
Vue Project Creation

- Frontend (vuejs)
- Backend (express)



Frontend and Backend Binding

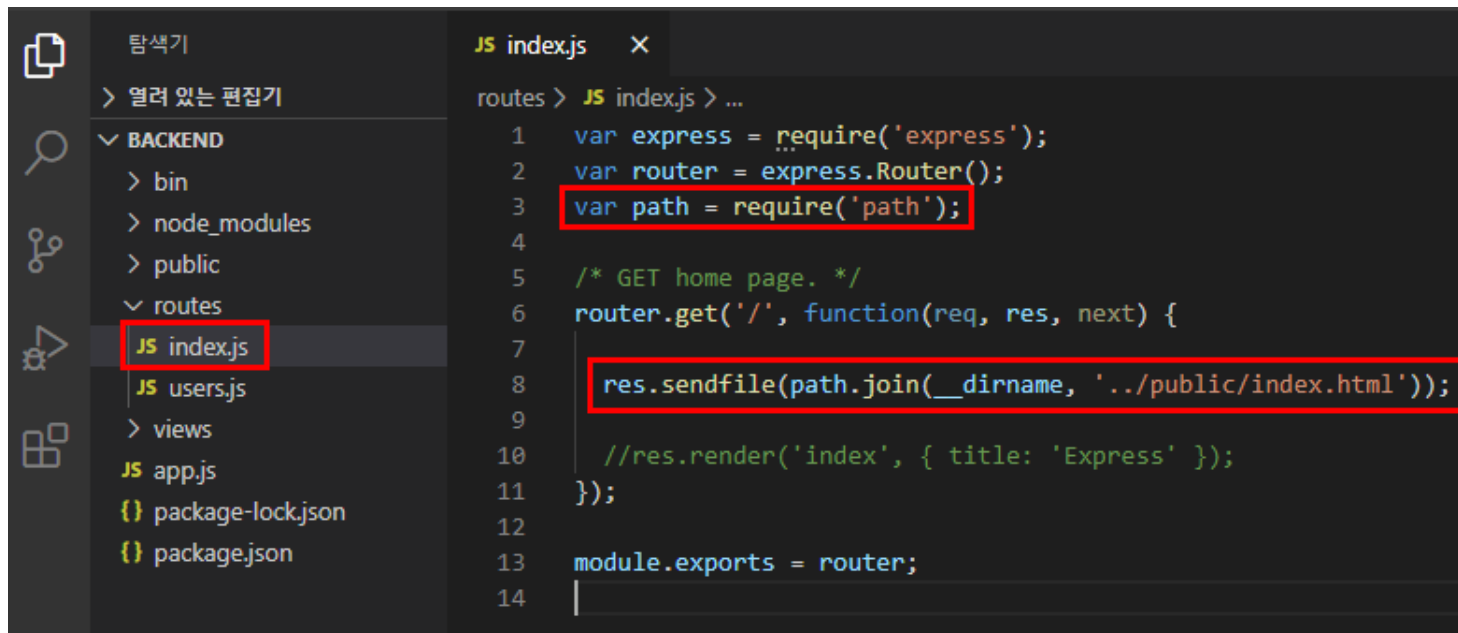
- Frontend (vuejs) – config/index.js
 - index: path.resolve(__dirname, '../../oneM2MApplicationBackend/public/index.html')
 - assetsRoot: path.resolve(__dirname, '../../oneM2MApplicationBackend/public')
 - npm run build



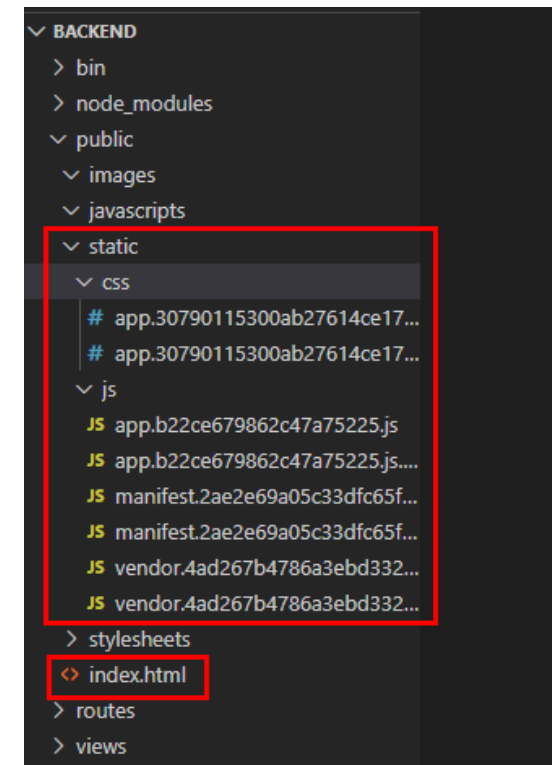
```
39 build: {
40   // Template for index.html
41   index: path.resolve(__dirname, '../../backend/public/index.html'),
42
43   // Paths
44   assetsRoot: path.resolve(__dirname, '../../backend/public'),
45   assetsSubDirectory: 'static',
46   assetsPublicPath: '/',
47
48   /**
49    * Source Maps
50    */
51
52   productionSourceMap: true,
53   // https://webpack.js.org/configuration/devtool/#production
54   devtool: '#source-map',
55
56   // Gzip off by default as many popular static hosts such as
57   // Surge or Netlify already gzip all static assets for you.
58   // Before setting to `true`, make sure to:
```


Frontend and Backend Binding

- Backend (express) – routes/index.js
 - `var path = require('path');`
 - `res.sendFile(path.join(__dirname, '../public/index.html'));`
 - `npm start`

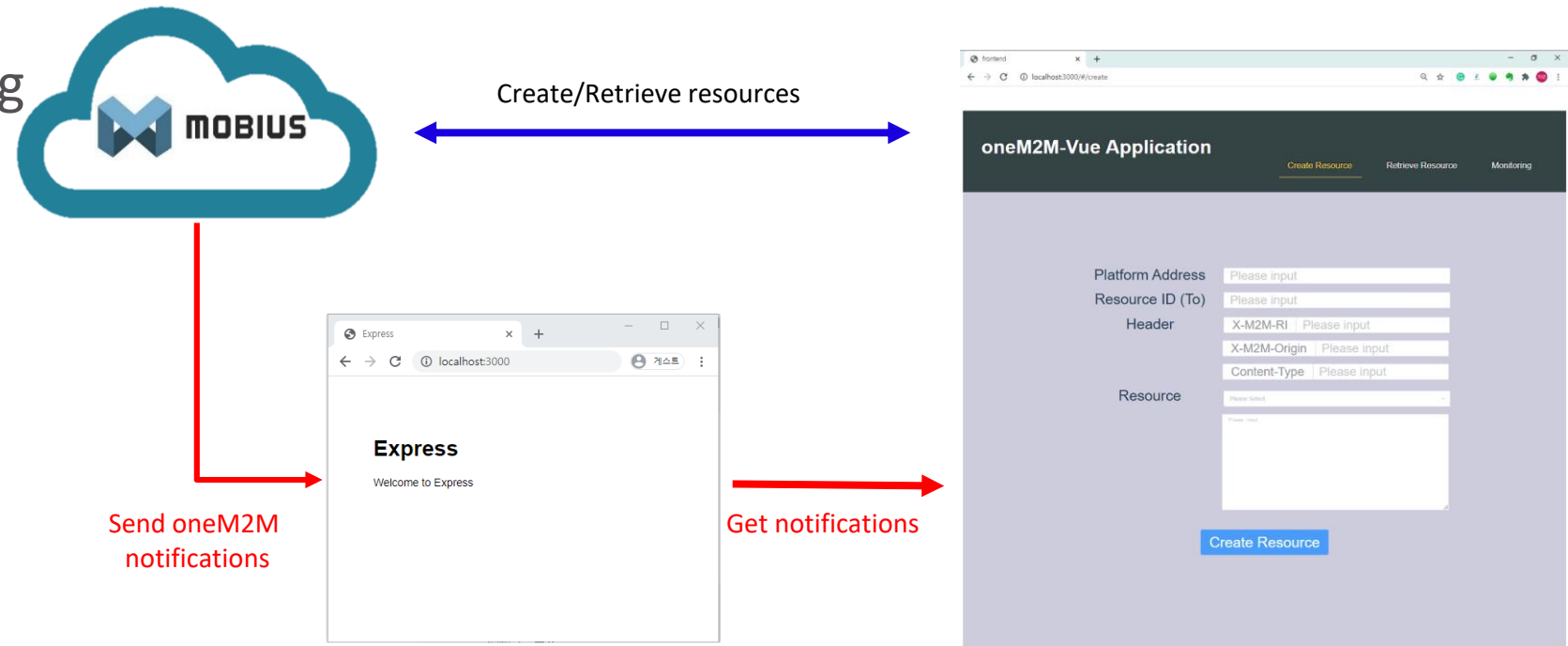


```
JS index.js X
routes > JS index.js > ...
1  var express = require('express');
2  var router = express.Router();
3  var path = require('path');
4
5  /* GET home page. */
6  router.get('/', function(req, res, next) {
7
8      res.sendFile(path.join(__dirname, '../public/index.html'));
9
10     //res.render('index', { title: 'Express' });
11 });
12
13 module.exports = router;
14
```



Web App Features

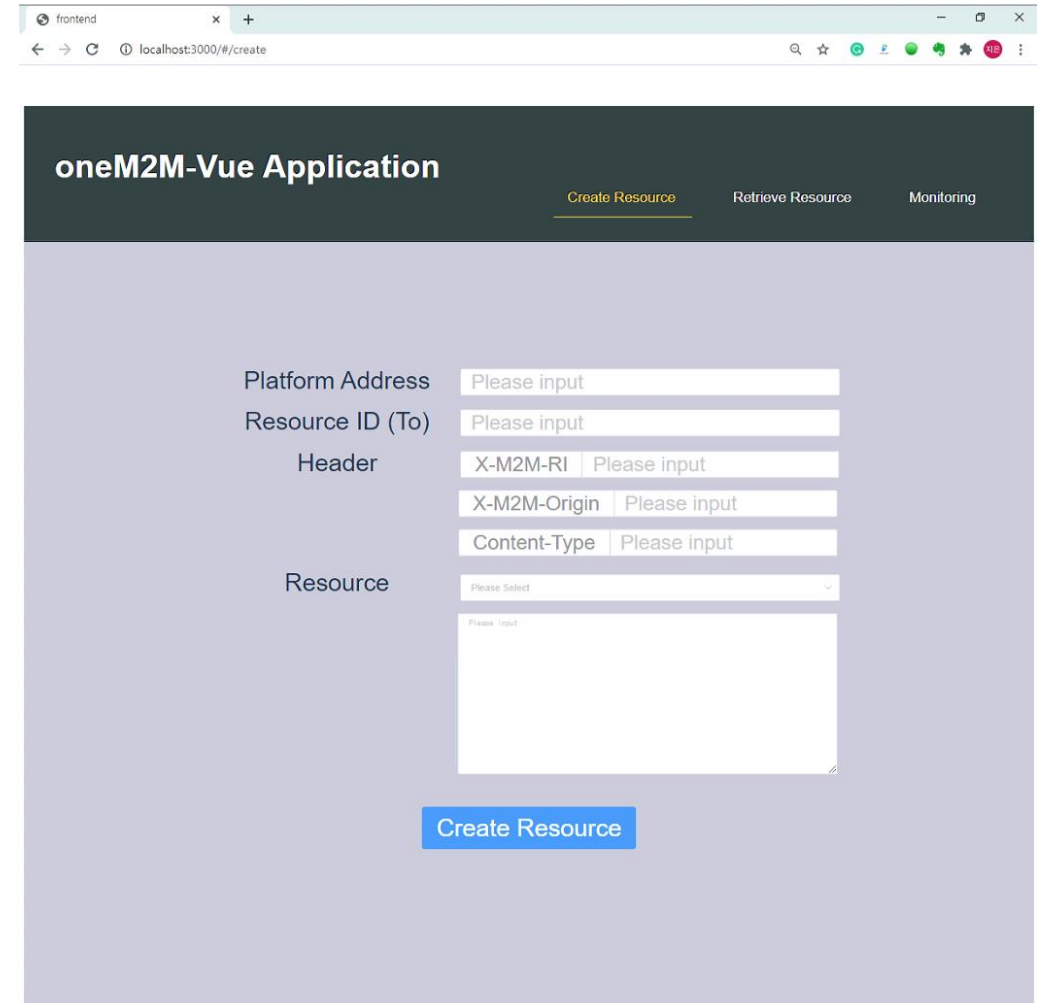
- Postman-like web app that also supports monitoring notifications using oneM2M APIs
 - Resource creation
 - Resource retrieval
 - Notification monitoring



Resource Creation

- oneM2M HTTP server address
- oneM2M request target
- oneM2M “X-M2M” HTTP headers
- HTTP POST with resource body

```
methods: {  
  create: function () {  
    const headers = {  
      "X-M2M-RI": this.Xm2mRI,  
      "X-M2M-Origin": this.Xm2mOrigin,  
      "Content-Type": this.contentType,  
    };  
    const url = "http://" + this.address + "/" + this.resourceId;  
    const body = this.resourceType;  
    axios.post(url, body, { headers }).then((response) => {  
      console.log(response.data);  
    });  
  },  
},  
},
```



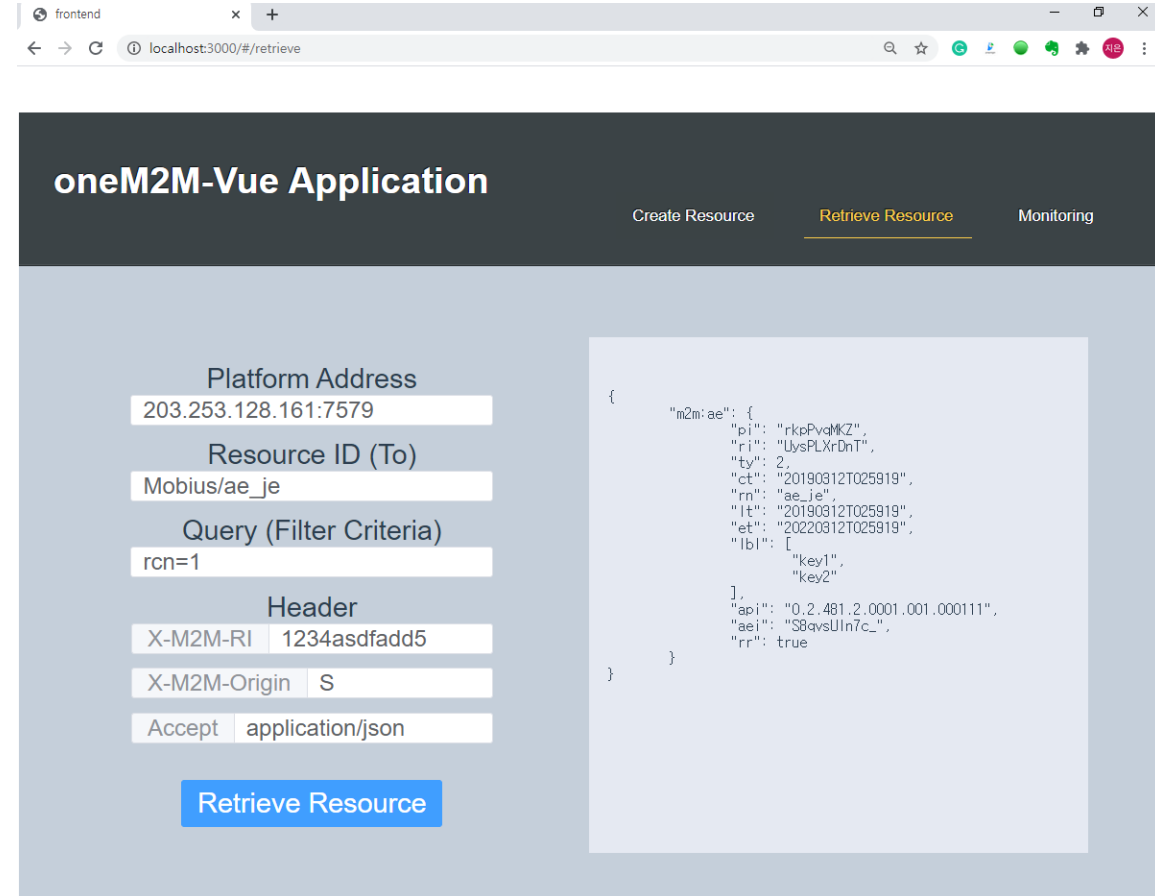
The screenshot shows a web browser window with the URL `localhost:3000/#/create`. The application title is "oneM2M-Vue Application". There are three tabs: "Create Resource" (active), "Retrieve Resource", and "Monitoring". The form contains the following fields:

- Platform Address:
- Resource ID (To):
- Header:
 - X-M2M-RI:
 - X-M2M-Origin:
 - Content-Type:
- Resource:
- A large text area for the resource body with the placeholder "Please input".
- A blue "Create Resource" button at the bottom.

Resource Retrieval

- oneM2M request target with filters
- oneM2M “X-M2M” HTTP headers
- HTTP GET
- Returned resource on the screen

```
methods: {  
  retrieve: function () {  
    const headers = {  
      "X-M2M-RI": this.Xm2mRI,  
      "X-M2M-Origin": this.Xm2mOrigin,  
      Accept: this.accept,  
    };  
    const url = "http://" + this.address + "/" + this.resourcdId + "?" + this.query;  
    axios.get(url, { headers }).then((response) => {  
      console.log(response.data);  
      this.response = response.data;  
    });  
  },  
},  
},
```



The screenshot shows the 'oneM2M-Vue Application' interface. The top navigation bar includes 'Create Resource', 'Retrieve Resource' (highlighted), and 'Monitoring'. The main form contains the following fields:

- Platform Address: 203.253.128.161:7579
- Resource ID (To): Mobius/ae_je
- Query (Filter Criteria): rcn=1
- Header: X-M2M-RI: 1234asfdadd5, X-M2M-Origin: S, Accept: application/json

A blue 'Retrieve Resource' button is at the bottom of the form. To the right, a light blue box displays the returned JSON data:

```
{  
  "m2m:ae": {  
    "pi": "rkpPvdMKZ",  
    "ri": "UysPLXrDnT",  
    "ty": 2,  
    "ct": "20190312T025919",  
    "rn": "ae_je",  
    "lt": "20190312T025919",  
    "et": "20220312T025919",  
    "bl": {  
      "key1",  
      "key2"  
    },  
    "api": "0.2.481.2.0001.001.000111",  
    "aej": "S8qvsUln7c_",  
    "rr": true  
  }  
}
```

Notification Monitoring

- Create a <subscription> resource first

Subscription CREATE Examples 0 BUILD

POST {{mp_url}}/{{cb}}/ae_je/cntje?rcn=1 Send

Params ☒ Authorization ☐ Headers (13) ☒ Body ☐ Pre-request Script ☐ Tests ☐ Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL ☒ JSON

```
1 {
2   "m2m:sub" : {
3     "rn": "sub12",
4     "enc": {
5       "net": [3]
6     },
7   },
8   "nu": ["http://localhost:12345"]
9 }
10 }
```

Notification Monitoring

- Receive a notification
 - when receives a notification from the platform, sends back the response
 - and then keep the last notification message
- Get the notification from the Backend
 - backend/routes/index.js
 - With HTTP GET request frontend gets the last notification and show on the screen

Notification Monitoring

- Create or Update resources to trigger event notification
 - refer the <subscription> resource

ContentInstance CREATE Examples 0 BUILD

POST `{{mp_url}}/{{cb}}/ae_ie/cntje` Send

Params Authorization Headers (13) **Body** Pre-request Script Tests Settings

☐ none ☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary ☐ GraphQL **JSON**

```
1 {
2   "m2m:cin": {
3     "con": "on"
4   }
5 }
```

- A notification message is sent by the oneM2M platform

```
{ 'm2m:sgn':
  { nev: { rep: [Object], sur: 'Mobius/ae_test/cnt_name/sub1', net: 1 } } }
```

Notification Monitoring

- Get the last notification from the backend, and print the notification message on the screen

```
mounted() {  
  axios.get("http://localhost:3000/notification").then((res) => {  
    console.log(res.data);  
    this.response = JSON.parse(res.data);  
  });  
},
```

