



Basic Vue JS 2 (หลักสูตรพื้นฐาน Vue.js)



ยุคสมัยนี้การพัฒนาเว็บแอปพลิเคชันส่วนใหญ่เน้นไปทางการทำเว็บที่เป็นแบบ Single Page Application (SPA) คือการทำงานจบในหน้าเดียว ไม่ต้องรีโหลด หรือเปลี่ยนหน้าไปมา จึงกำเนิด JavaScript Framework ที่รองรับการทำงานลักษณะนี้มากมาย ไม่ว่าจะเป็น Angular, React, NodeJS และ Vue.js ก็เป็นหนึ่งใน JavaScript Framework ที่กำลังได้รับความนิยมใช้กัน

ซึ่งในหลักสูตรนี้จะแนะนำภาพรวมตั้งแต่เริ่มต้นการใช้งาน ไปจนสามารถประยุกต์พัฒนาโปรเจกต์ของผู้เรียนได้เอง จึงเหมาะสำหรับผู้สนใจเกี่ยวกับการพัฒนาเว็บแอปสมัยใหม่ด้วย Vue.js JavaScript framework ยอดนิยมนำมาใช้

วัตถุประสงค์:

- เข้าใจแนวคิดของ Component Based Application ใน Vue.js
- เรียนรู้และเข้าใจ การ Setup Vue Instance
- ใช้งาน Basic Directives ได้
- จัดการ State ได้
- จัดการ Event Handler ได้
- จัดการ Attribute Binding ได้
- สามารถจัดการฟอร์ม Forms and Form Validation โดยใช้ vee-validate
- เข้าใจ Life Cycle Method
- สามารถเชื่อมต่อกับ HTTP Service ได้
- ประยุกต์สร้าง Single page applications (SPA) ได้

กลุ่มเป้าหมาย:

- นักเรียนนักศึกษา
- ครู อาจารย์ วิทยากรที่สนใจ
- นักวิชาการ นักไอที หรือผู้ดูแลระบบ
- ตลอดจนผู้สนใจทั่วไป



ความรู้พื้นฐาน:

- พื้นฐาน HTML & CSS มาบ้าง
- พื้นฐาน JavaScript มาบ้าง
- พื้นฐานการใช้งาน Windows
- พื้นฐานการค้นหาข้อมูลจากอินเทอร์เน็ต

ระยะเวลาในการอบรม:

- 18 ชั่วโมง (3 วัน)

ราคาคอร์สอบรม:

- 6,000 บาท / คน (ราคานี้ยังไม่ได้รวมภาษีมูลค่าเพิ่ม)

วิทยากรผู้สอน:

- อาจารย์สามิตร โกยม

คอร์สที่ควรอบรมก่อนหน้า:

- Basic JavaScript

คอร์สต่อเนื่องที่แนะนำ:

- ไม่มี

เนื้อหาการอบรม:

Module 1: Getting Started

- Hello World
- Introduction to Vue.js
- Setting up a local development environment (the easy way)



Module 2: Fundamentals of Vue.js

- Introduction to this section
- Working with templates
- Methods
- A word about ES6 arrow functions
- Introduction to directives
- Exercises: String interpolation, methods and directives
- Using expressions with directives
- Introduction to events
- Passing arguments to event listeners
- Directive & event modifiers
- Key modifiers
- Modifier keys
- Exercises: Events & modifiers
- Two-way data binding
- Security: Outputting and escaping HTML
- Rendering elements only once
- Conditionally rendering elements
- Showing and hiding elements
- Hiding elements until the Vue instance is ready
- Looping through arrays
- Accessing the loop index
- Looping through object properties
- Looping through number ranges
- Understanding the v-for update strategy
- Array change detection
- Exercises: Two-way data binding, conditionals, and loops
- Optimizing performance with computed properties
- Adding getters & setters to computed properties
- Watchers



- Exercises: Computed properties & watchers
- Filters
- Exercises: Filters
- Styling with inline CSS styles
- Styling with CSS classes
- Exercises: Styling with inline styles and classes
- Shorthands for events and bindings

Module 3: Deep dive: The Vue Instance

- Introduction to this section
- Accessing a Vue instance outside of its declaration
- Using multiple Vue instances on the same page
- Proxying
- Understanding reactivity
- Asynchronous update queue
- Understanding the Virtual DOM
- Adding watchers dynamically
- Accessing the DOM with \$refs
- Mounting templates dynamically
- Using inline templates
- Destroying a Vue instance
- Vue instance lifecycle & hooks
- Using lifecycle hooks

Module 4: Setting up a Development Environment with Webpack & Vue CLI

- Introduction to this section
- Installing Chrome developer tools extension
- Introduction to Vue CLI
- Creating a project with Vue CLI
- Understanding the project structure



- Single file components
- Building for production

Module 5: Components

- Introduction to components
- Why the data property must be a function
- Global & Local Components
- Component naming conventions
- Creating a component in the project
- Exercises: Global, local and child components
- Organizing components
- Global & scoped styles
- Passing data to components
- Validating passed data
- Working with events
- Communicating through an event bus
- Exercises: Passing data & events
- Slots
- Named slots
- Dynamic components
- Keeping dynamic components alive
- Dynamic components lifecycle hooks
- Vue developer tool in action

Module 6: Mixins & Filters

- Introduction to mixins
- Using mixins
- How mixins are merged in
- Global mixins
- Global filters



Module 7: Forms

- Full source code available within GitHub repository
- Text inputs & textareas
- Checkboxes
- Radio buttons
- Selects (dropdowns)
- Modifiers
- How the v-model directive works
- Adding default values
- Submitting forms '

Module 8: Animations & Transitions

- Introduction to transitions and animations
- Understanding single element transitions
- Transitioning with CSS classes
- Implementing our first transition
- Specifying transition names
- Specifying custom transition classes
- Implementing a custom CSS animation
- Mixing transitions and animations
- Transitioning between elements
- Transition modes
- Transitioning elements initially (on page load)
- Transitioning with JavaScript hooks
- Ignoring CSS classes
- Transitioning dynamic components
- Transitioning multiple elements
- Transitioning moving elements



Module 9: Routing (SPA)

- Full source code available within GitHub repository
- Introduction to Single Page Applications (SPA)
- Installing vue-router
- Enabling the router
- Registering routes
- Rendering routed components
- Changing the router mode
- Catch-all route
- Moving routes to file
- Adding navigation links
- Styling the active link
- Dynamic route matching
- Linking to dynamic routes
- Named routes
- Retrieving route parameters
- Using route props
- Reacting to parameter changes
- Navigating programmatically
- Navigating in the browser's history stack
- Redirecting
- Aliases
- Nested routes
- Query parameters
- Hash fragment
- Controlling the scroll behavior
- Named views
- Route transitions
- Route guards
- Component guards



- Global guards & meta fields
- Reacting to navigations

Module 10: Connecting to Servers through HTTP

- Introduction to using HTTP in Vue
- Setting up vue-resource
- Fetching data with GET requests
- URI templates
- POST requests
- Using resources
- Custom resource actions
- Global configuration
- Configuration within components
- Interceptors

Module 11: Vuex

- Introduction to Vuex
- Why Vuex is needed
- Installing Vuex
- Setting up a simple store
- Accessing a store's state
- Changing state
- Introduction to getters
- Implementing a getter
- Accessing other getters
- Passing arguments to getters
- Mapping getters to computed properties
- Introduction to mutations
- Implementing a mutation
- Committing mutations with payloads



- Mapping mutations to methods
- Using constants for mutation types
- Two-way data binding
- Introduction to modules
- Implementing modules
- Accessing root state from getters
- Accessing root state from actions
- Adding mutations to modules
- Introduction to namespaces
- Adding namespaces
- Accessing namespaced types with helpers
- Accessing root getters from namespaced modules
- Dispatching root actions and committing root mutations
- Integration with developer tools
- Should you use Vuex?

Module 12: Vue Live workshop

E-commerce workshop

- Displaying products
- Adding products to the cart
- Displaying the cart info in the menu
- Switching between views
- Displaying the cart
- Handling adding products already in cart
- Increasing and decreasing product quantities
- Checking out

Mail application workshop

- Introduction to the application



- Creating components for the views
- Adding the sidebar markup
- Switching between views
- Loading messages from file
- Adding numbers to navigation menu
- Displaying the inbox
- Marking messages as important
- Displaying sent, important and trashed messages
- Displaying a message
- Automatically marking messages as read
- Navigating back to previous views
- Deleting messages
- Marking messages as read or unread
- Refreshing the inbox
- Sending messages