Week 5: Vue.js Framework

**Task 1: Components Basics**

***Reflection:***

Components are part of websites or web apps, which are reusable. Change of data in one component does not change data in another component. Components have name.

Components are piece of the website made up of Vue. Each components have block of HTML, CSS and JavaScript, to encapsulate function.

Blocks of HTML, CSS and JavaScript which the component have is called template. A template bind’s the data of the component to the DOM in the template. One component can be used in another component to enhance feature of the website.

Components may have child components or a parent component. Parent component can pass data by props, and child components pass data by events.

**Task 2: Understanding Handling User Input**

***Reflection:***

In Vue a directive called v-model is used to create two-way data binding on form input, text area, and select elements. Based on the input type, it automatically picks the correct way to update the element,

V-model is used to get data from a form field and updating the field. Not all the fields can be used by v-model. initial value, Checked or selected attributes found on the form elements. Initial values are to be declared in the JavaScript side inside the data option of the component. It uses data option for the component as truth.

The improvement on Vue over HTML built-in input type is not always sufficient to meet need. Vue components creates reusable inputs with completely customised behaviour.

**Task 3: Learning Component Registration**

***Reflection:***

A component needs to be registered before use. A component also needs a name before registration. Naming depends on the component’s extent. Whether it is local or global. When using the component directly in the DOM, it is strongly recommended to use W3C rules, to avoid conflicts with current and future HTML elements.

When the components are globally registered, they can be used in the template in the root Vue instance created after registration. Globally registered components are also available to each other.

Global registration of a component makes it available even when it is not in use. Globally registered components are included with the final build, So it unnecessarily increases the amount of JavaScript downloaded. Locally registered components are not available in subcomponents.

**Task 4: Learning Props, Events and Slots**

***Reflection:***

Props:

Props are used to pass data. Props are case sensitive. When a template is used in a DOM, the naming changes according to the HTML naming convention.

When using String template this limit does not apply. Props can pass static and dynamic value. Props can pass Array, objects, properties of an object, Boolean, number, and string.

Data binding happens in one way direction. If a prop change it’s value in the parent component, the change is reflected in the child component. But it does not happen in the opposite direction. That is if there is a change of a prop which the parent also has, happens inside child, it will not affect the prop inside the parent. Data type of a prop is validated.

Slots:

Slot is a content distribution API. Slots distributes template and other component. Slot has access to the same instance properties as the template. We can use data inside a slot.

Reference:

[Slots — Vue.js (vuejs.org)](https://v2.vuejs.org/v2/guide/components-slots.html) Accessed: 8/09/2022.

[Component Registration — Vue.js (vuejs.org)](https://v2.vuejs.org/v2/guide/components-registration.html) Accessed: 8/09/2022.

[Form Input Bindings — Vue.js (vuejs.org)](https://v2.vuejs.org/v2/guide/forms.html) Accessed: 8/09/2022.

[Components Basics — Vue.js (vuejs.org)](https://v2.vuejs.org/v2/guide/components.html) Accessed: 8/09/2022.