**Vue.js**   
  
Easy to understand syntax and logic   
Vue JS uses JSX alongside of HTML elements. The syntx and layout for Vue components are easy to understand as it follows the template of making a webpage. The template serves as a directive Vue to produce the final markup elements and how it should be displayed based on its internal state. The state changes and properties reside in the script tag and the scoped css resides in the style tags. It is easy to understand a Vue component because it is laid out like a webpage with HTML as the mark up of the webpage, Javascript handing the logic and CSS handling the styling.   
  
Components as objects   
Vue sees components as objects and registers that component globally. Vue components do not need to extend. To assign property with a new value, you simply assign it to a new value. All properties of the object are accessible with the “this” keyword with no need to bind it within the constructor method. You define and bind the methods by creating functions in the methods object.   
  
  
**Vue vs React**   
  
Vue is a framework, React is a library  
Most UI based applications need routing and state management. Routing handles HTTP requests while state management presides over state changes within components.   
  
Vue as a framework offers first party complete solutions with routing with Vuex and state management with Vuex. Compared to React, it needs to use third party libraries such as Redux for state changes and React Routing for routing.   
  
  
Styling   
Vue has built in support for styling even scoped styling for components. Vue also incorporates dynamic class names through V directives and the classNames library. With React, you would have to find the library and install it. Vue also auto prefix CSS styles.