	- new lines
	- updated lines

	Vue	
Version used	1.0.26	
Language used	ES6	
Templating logic		
	Vue	
Text interpolation	{{ message }}	
Transform interpolation	{{ message | filter }}	
Transform with arguments	{{ message | filter 'arg1' arg2 }}	
Bind text content		
Bind dangerous html Content by directive		
Bind href	<a v-bind:href="url"> <a v-link="url"> With official vue-router plugin is the best way, I think.	But we can use shorthand syntax like <a :href="url"> Anyway, both methods are not recommended.
Dangerous raw HTML output	{{{ raw_html }}}	
Bind attribute value		
DOM add/remove	 And 	
DOM show/hide	 And 	
Repeat	 	
Bind event handler	<button v-on:click="clicked"> Or just <button @click="clicked"> <form @submit.stop.prevent="submit"> <input @keyup.enter="save">	
Components		
	Vue	
Define component ES6	class Polyglot extends Vue	Also you can define collections of components directly in property «components» of vue instance, using ES6 syntax {Copm1,Comp2,Comp3}
Define component ES5	Vue.extend({}) Vue.component(name, {})	
Component communication		

	Vue	
One-way data binding	<child :foo='bar' />	
String literal prop	<child foo='bar' />	
Two-way binding (native)	<input v-model="foo" />	
Two-way binding (components)	<child :foo.sync="bar"></child>	
One-time binding	<child :foo.once="bar" />	
Component internal state		
	Vue	

Set initial state	data: {
	foo: 1
	}
	data: {
	return {
	model: {},
	collection: []
	}
	}
		If you need to set nested data values,
		you have to use function returning value

Set state	this.foo = 2	
Property validation		
	Vue	
Prop validation key	{props: {}}	

Required props without type	{email: {
	required: true
	}
	}	

Prop validate Number	{myNum: Number}	
Prop validate String	{myStr: String}	
Prop validate Array	{myArr: Array}	
Prop validate Boolean	{myBool: Boolean}	
Prop validate Function	{myFunc: Function}	Now supported
Prop validate Object	{myObj: Object}	
Prop validate Date	{myDate: Date}	It`s not documented, but I have some experience with this feature and don`t have any errors
Prop validate union	{myVar: [String, Number]}	
Component lifecycle methods		
	Vue	
Initialized	created	
Before compile template	beforeCompile	
After compile template	compiled	
DOM Ready	ready	
On prop change	none	
Component updated	none	
Before destroy	beforeDestroy	
After destroy	destroyed	
