Plugin Vue
Introduced in Tiki21, this wiki plugin allows you to add a Vue.js Use this plugin to integrate powerful Vue.js features seamlessly into your Tiki pages.

## **Parameters**

Plugin PluginTikiDocFromCode failed: Either the module, plugin, preferences, tracker field or prefsexport parameter must be set, but not two or more.

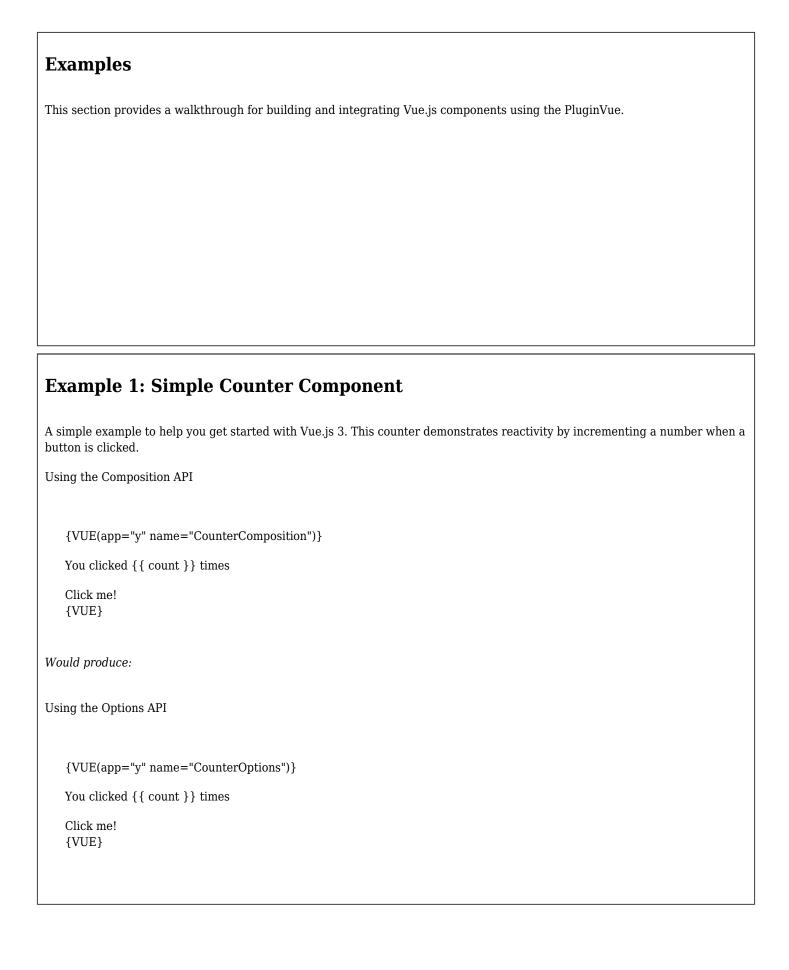
## **Notes**

- \*\*API Support\*\*: You can now write Vue.js components using either the Composition API (recommended for modern development) or the Options API (ideal for those migrating older components or new to Vue.js).
- \*\*Scoped Styles\*\*: Styles defined in components will not leak into the global CSS scope, making it easier to maintain and debug your projects.
- \*\*Reactive State Management\*\*: Leverage Vue's reactivity system to build highly interactive and dynamic interfaces.

## 'Prerequisites':

The rest of this documentation assumes that you have basic familiarity with Vue.js. If you are completely new to Vue.js, it's recommended to first learn the framework's fundamentals, such as component structure, reactivity, and lifecycle methods, before using this plugin. Prior experience with other frameworks is not required, but having a good understanding of Vue.js concepts will help you make the most of the PluginVue.

Once you've grasped the basics, you'll find it easier to build and integrate Vue.js components into your Tiki site.



Example 2: TODO App (bootstrap themed)
A more advanced example that demonstrates state management, event handling, and dynamic rendering with Vue.js. This application allows users to add, complete, and delete tasks.
Using the Composition API
{VUE(app="y" name="TodoApp")}
Todo MVC App
You have ({{ todoCount }}) todos
Toggle All
• □ {{ todo.text }}
Delete
Double click on a task to edit
Completed Todos: {{ completedCount }}
{VUE}
Would produce:
Resources

• https://vuejs.org/guide/introduction.html