These are other notes on View from the Reaction-Timer Project  
Graphical user interface, website

Description automatically generated  
  
  
GitHub Repo:  
<https://github.com/lionel5116/VueReactionTimerProject.git>  
  
**Tip**: Another tip on emitting methods  
**Tip**: Passing props (just a refresher)  
**Tip**: Adding Bootstrap to Vue  
Tip: Adding Routing in Vue 3  
Tip: Refactoring the app for the routing and nav  
Tip: Deployment and routing  
Tip: Vuex (State Management) in Vue (And the differences from REDUX)  
Tip: Using axios in vue  
Tip: Running unit tests with jest an vue tools  
Tip: Removing the white space above the navigation  
Tip: Navigation in Vue (to another component with a button click)  
Tip: Bootstrap Form (Bootstrap 5)  
Tip: Building a bootstrap table from scratch with regular bootstrap  
Tip: Adding the backend to your project (for MONGO DB Integration)  
Tip: Writing a record to MongoDB Atlas from VueJs Application /Bootstrap, binding etc..  
  
  
**Tip**: Another tip on emitting methods  
If you look below, we are emitting a method that we will name ‘end’ in the parent component. Notice how we are passing data along with it.  
Text

Description automatically generated  
Now over in our parent component:  
Notice below how we are making reference to the emitter by its name (‘end’)  
We are pointing it to the method “endgame”  
In the method, we are specifying an argument as a parameter for the function. When created the emitter, we passed in some data, so what that means is that the method create automatically receives that data as it’s parameter.  
Text

Description automatically generated  
  
**Tip**: Passing props (just a refresher)  
We declare props

A screenshot of a computer

Description automatically generated with medium confidence  
  
Then to receive the prop from the parent:

Text

Description automatically generated  
**Tip**: Adding Bootstrap to Vue  
In vue 3 , you have to use bootstrap 5  
You can install bootstrap the same way (regular npm) as any other node.js project  
https://getbootstrap.com/docs/5.0/components/navbar/

npm i bootstrap@5.0.1

npm install --save @popperjs/core

Then In main.js

import "bootstrap/dist/css/bootstrap.min.css"

import "bootstrap"  
  
Create a nav component

Text

Description automatically generated

Text

Description automatically generated  
  
Graphical user interface, application, website

Description automatically generated  
  
You can then use bootstrap anywhere in your application. Just refer to the bootstrap docs  
Graphical user interface, text, application

Description automatically generated  
Text

Description automatically generated  
  
Graphical user interface, text, application

Description automatically generated  
  
Tip: Adding routing in Vue 3  
https://www.vuemastery.com/blog/vue-router-a-tutorial-for-vue-3/  
Text

Description automatically generated  
  
In main.js

A screenshot of a computer

Description automatically generated with medium confidence  
  
  
In your Custom Nav Component

Text

Description automatically generated  
  
Graphical user interface, application

Description automatically generated  
  
Graphical user interface, application

Description automatically generated  
  
For the home link

Text

Description automatically generated  
  
You don’t need a route  
Graphical user interface, text, application, chat or text message, website

Description automatically generated  
Graphical user interface, website

Description automatically generated  
  
Tip: Refactoring the app for the routing and nav  
I refactored the app to correct the navigation  
Text

Description automatically generated  
  
Text

Description automatically generated  
  
Text

Description automatically generated  
  
Graphical user interface, text, application, chat or text message

Description automatically generated  
  
Graphical user interface

Description automatically generated with low confidence  
  
Background pattern

Description automatically generated

Tip: Deployment and routing  
When I deployed the app, the routing worked  
  
Graphical user interface, application

Description automatically generated  
  
Also the addition to vue.config also fixed the deployment issue with the paths as well  
A screenshot of a computer

Description automatically generated with medium confidence  
  
Tip: Vuex (State Management) in Vue (And the differences from REDUX)  
This came from this video  
<https://www.youtube.com/watch?v=nFh7-HfODYY>  
  
To add vuex to your project  
npm install vuex@next –save  
  
Then we can create our store  
Text

Description automatically generated  
  
To add our store, we make a reference in our main.js file

Text

Description automatically generated  
  
Now every component in our application has access to our store  
  
We will demonstrate two types of actions in our store

Text

Description automatically generated  
  
commit: allows to fire an action in your store  
dispatch: allows you to fire an action in our store  
To fetch data from our store we use the $store.state.<property>  
  
Text

Description automatically generated  
Like stated above to change some state in our store, we can use the mutations section to fire a method the update state.  
The actions section allows us to fire asynchronous code (api) and change a state’s property  
(Tip: Using axios in vue)  
npm install axios  
  
Differences between Vuex and REDUX  
  
Setup:  
Vuex:  
The setup for Vuex is really quite simple:  
You create a store using the createStore from vuex. In the index.js file where you create the store, you create you actions, mutations, getters and setters. Then in your main.js file, when you are issuing the creatApp command, you use the .use(store) command. After this, your entire application has access to the store.  
  
Redux:  
Redux is a different beast, it requires a lot more wireup and configuration to get it work. Even referencing the state and consuming it is more code intensive.  
  
Changing State:  
Vuex:  
The only way to change state in your app is by using mutations  
<https://vuex.vuejs.org/guide/mutations.html>

Redux:  
You used **Reducers** to return a brand new instance (part of the overall state tree)  
  
Actions:

<https://vuex.vuejs.org/guide/actions.html>  
Vuex:  
You create actions as part of the store’s index.js file. Actions also allow you to call asynchronous methods inside of them as well.

Redux:  
Similar, but you mainly create actions in a separate actions file.

Tip: Running unit tests with jest an vue tools  
Website for unit test examples  
<https://v2.vuejs.org/v2/cookbook/unit-testing-vue-components.html?#Why-test>  
  
Run these commands to set up unit testing

<https://v1.test-utils.vuejs.org/installation/#using-vue-test-utils-with-jest-recommended>

vue add unit-jest

npm install --save-dev @vue/test-utils

I created a component to unit test  
Text

Description automatically generated  
  
I added some test code  
Text

Description automatically generated  
  
Ran the test command to test one component

npm run test:unit foounit  
  
Text

Description automatically generated

Tip: Removing the white space above the navigation  
<https://stackoverflow.com/questions/58583789/how-to-remove-white-space-over-the-bootstrap-navigation-bar-in-vue-js>

The default styling had a margin-top: 60px

One you remove that:

Text

Description automatically generated  
  
Graphical user interface, application, website

Description automatically generated

Tip: Navigation in Vue (to another component with a button click)  
<https://router.vuejs.org/guide/essentials/navigation.html#navigate-to-a-different-location>  
Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated  
  
Graphical user interface, text, application

Description automatically generated

Tip: Bootstrap Form (Bootstrap 5)  
https://getbootstrap.com/docs/5.0/forms/layout/  
Look here for form layout options  
For Form Validation (out of the box input field validation) :  
https://getbootstrap.com/docs/5.0/forms/validation/  
  
  
For rows and columns:  
Graphical user interface, text, application, email

Description automatically generated  
  
For a form:  
Graphical user interface, text, application, email

Description automatically generated  
  
Graphical user interface, text, application, email

Description automatically generated  
  
Tip: Building a bootstrap table from scratch with regular bootstrap  
This example came from this article  
https://blog.logrocket.com/table-component-from-scratch-vue-3-bootstrap/

In the end, you do get this:

Graphical user interface

Description automatically generated  
  
Components:  
Text

Description automatically generated  
  
Using Static Data  
  
Using a Database – API Call  
Text

Description automatically generated  
  
A screenshot of a computer screen

Description automatically generated with medium confidence  
  
Graphical user interface

Description automatically generated with medium confidence

Tip: Adding the backend to your project (for MONGO DB Integration)  
Added a backend to my project  
Text

Description automatically generated  
  
You need to run   
npm init   
when you create folder structure  
  
Also install the following packages below:

Text

Description automatically generated  
  
I used the same wire-up for mongo db and express as the MERN project from the Udemy course  
Graphical user interface, text, application, email

Description automatically generated  
  
And it works

Graphical user interface, text, application, email

Description automatically generated  
  
Graphical user interface, application

Description automatically generated  
  
Text

Description automatically generated

<http://localhost:5000/api/clients/createClientRecord>  
POST METHOD

 {

  "ClientCode":"LIONEL5116",

    "LastName":"Jones",

    "FirstName":"Lionel",

    "Address":"16543 Magnolia Dr",

    "City":"Houston",

    "State":"Texas",

    "Zip":"77063",

    "Phone":"714-098-9865",

    "Phone2":"715-098-7612",

    "Cell":"854-098-4321",

    "Notes":"Adding a test record for Mongo DB Adding a client record for HuckeClient",

     "Email":"lionel.jones@gmail.com"

  }

Writing the record from VueJs Project

Tip: Writing a record to MongoDB Atlas from VueJs Application /Bootstrap, binding etc..

A picture containing text, screenshot, monitor, indoor

Description automatically generated  
  
And It WORKS!!!!!!!

A screenshot of a computer

Description automatically generated

Text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated  
  
Text

Description automatically generated