1. Cấu trúc thư mục
   * 1. Tham khảo nuxt
2. Component
   * 1. Common -- cm-name
     2. Domain – cd-name
     3. Pages -cp-name
3. Cách đặt tên:
   * 1. theo key dễ kiếm
     2. không trùng lặp
     3. không đổi tên khác với tên file
4. Cấu trúc ts
   * 1. Hướng thuần ts
     2. Hướng setup
     3. Hướng componentApi
5. Cấu trúc chia component
   * 1. Layout -> page -> component page -> component common
6. Constant:
   * 1. Không gán cứng
7. Pinia,vs vuex
8. Css, vuetify
9. Global property
10. Ultil, plugin
11. Lodash custom
12. Unit test
13. Quy tắc
    * 1. comment
14. Thư viện
    * 1. Husky
      2. Loadash
      3. Mathlive

Axios

Meta route

<route lang="yaml">

meta:

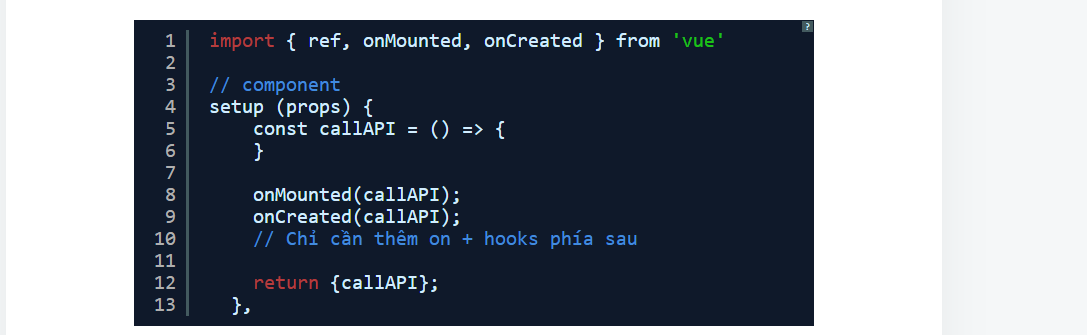
  layout: blank

  action: read

  subject: Auth

</route>

1. Thuộc tính layout trong vue 3, Route, meta => win
2. Kiểm tra khi nào dung optional và composition api, Typescript=> bach (win suport)
3. Tích hợp vuetify và css tích hợp vào component=> sinh (win suport)
4. Pinia,vs vuex=> win
5. Ultil, plugin, Axios, Vue 3=> sinh (win suport)
6. Lodash custom, Global property=> bach (win suport)

<script lang="ts" setup>

import { useLayouts } from '@layouts'

*interface* Props {

  popperInlineEnd?: *boolean*

  tag?: *string*

  contentContainerTag?: *string*

  isRtl?: *boolean*

}

*const* props = withDefaults(defineProps<Props>(), {

  popperInlineEnd: false,

  tag: 'div',

  contentContainerTag: 'div',

  isRTL: false,

})

*const* refPopperContainer = ref<*HTMLElement*>()

*const* refPopper = ref<*HTMLElement*>({

  left: '0px',

  top: '0px',

}

)

onMounted(updatePopper)

watch([isAppRtl, appContentWidth], updatePopper)

*const* selectAllEmailCheckbox = computed(

  () *=>* store.emails.length && store.emails.length === selectedEmails.value.length,

)

*const* updatePopper = async () *=>* {

*const* { x, y } = await computePosition(refPopperContainer.value, refPopper.value, {

    placement: props.popperInlineEnd ? (props.isRtl ? 'left-start' : 'right-start') : (props.isRtl ? 'bottom-end' : 'bottom-start'),

    middleware: [

      // eslint-disable-next-line @typescript-eslint/no-non-null-assertion

      flip({ boundary: document.querySelector('body')! }),

      // eslint-disable-next-line @typescript-eslint/no-non-null-assertion

      shift({ boundary: document.querySelector('body')! }),

    ],

  })

  popperContentStyles.value.left = `${x}px`

  popperContentStyles.value.top = `${y}px`

}