

MMISP Frontend Documentation

Design Phase

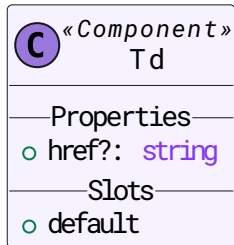
Table of contents

1. Components	4
1.1 Td	4
1.2 Table	4
1.3 Th	4
1.4 DynTable	5
1.5 Info	5
1.6 Pill	6
1.7 DistributionPill	7
1.8 PillCollection	7
1.9 Boolean	8
1.10 PillNavigation	8
1.11 Button	9
1.12 Card	9
1.13 CardRow	10
1.14 Checkbox	10
1.15 Select	11
1.16 Input	12
1.17 SideMenuDivider	13
1.18 SideMenuEntry	13
1.19 SideMenu	14
1.20 TopMenu	15
1.21 Layout	16
1.22 DatePill	16
1.23 SettingsEntry	17
1.24 CustomNode	17
2. Pages	19
2.1 /	19
2.2 /admin/keys	19
2.3 /admin/keys/[id]/edit	19
2.4 /admin/keys/[id]/view	19
2.5 /admin/keys/list	19
2.6 /admin/servers	20
2.7 /admin/servers/[id]/edit	20
2.8 /admin/servers/[id]/view	20
2.9 /admin/servers/list	20

2.10	/admin/users	20
2.11	/admin/users/[id]/edit	21
2.12	/admin/users/[id]/view	21
2.13	/admin/users/list	21
2.14	/event	21
2.15	/event/[id]/edit	21
2.16	/event/[id]/view	21
2.17	/event/list	22
2.18	/galaxy	22
2.19	/galaxy/[id]/edit	22
2.20	/galaxy/[id]/view	22
2.21	/galaxy/list	22
2.22	/settings	23
2.23	/tags	23
2.24	/tags/[id]/edit	23
2.25	/tags/[id]/view	23
2.26	/tags/list	23
2.27	/workflow/modules	24
2.28	/workflow/modules/[id]/edit	24
2.29	/workflow/modules/[id]/view	24
2.30	/workflow/modules/list	24
2.31	/workflow/trigger	24
2.32	/workflow/trigger/[id]/edit	25
2.33	/workflow/trigger/[id]/view	25
2.34	/workflow/trigger/list	25
3.	Layouts	26
3.1	/	26
4.	Error Pages	27
4.1	/	27

1. Components

1.1 Td



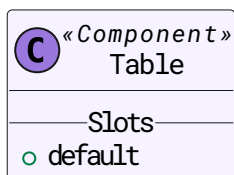
1.1.1 Props

href: `string` | `undefined`

1.1.2 Slots

default

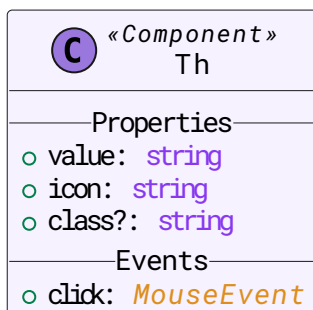
1.2 Table



1.2.1 Slots

default

1.3 Th



icon: string

class: string | undefined

click: MouseEvent

1.4 DynTable

1.4.1 Props

```
T extends readonly HeaderEntry[]
```

header: T

```
data: FlatUnion<MapNameToDisplayComp<T>>[]
```

```
urlCb: ((id: string) => string | undefined) | undefined
```

1.5 Info

Displays a text with a default background color of surface1. Also sets the default padding and border radius. You can override this by passing your own classes.

1.5.1 Props

```
text: string | undefined
```

The text to be displayed.

`class: string | undefined`

Additional classes to be applied.

1.5.2 Slots

default

1.6 Pill

C

«Component»

Pill

Properties

○ label?: string

○ text?: string

○ icon?: string

○ class?: string

○ style?: string

Slots

○ default

1.6.1 Props

label: string | undefined

text: string | undefined

icon: string | undefined

class: string | undefined

style: string | undefined

1.6.2 Slots

default

1.7 DistributionPill

<div><div><div>C</div></div><div>«Component» DistributionPill</div></div>	
Converts a distribution number to the equivalent text. Uses the { <code>@link Pill</code> } component to display the text and icon	
Properties	
<div>The distribution number of the the event.</div> <div><div>- 0: Your organization only</div><div>- 1: This community only</div><div>- 2: Connected communities</div><div>- 3: All communities</div><div>- 4: Sharing group</div><div>- 5: Inherit Event</div></div> <div><code>distribution?: number</code></div>	

Converts a distribution number to the equivalent text. Uses the `Pill` component to display the text and icon

1.7.1 Props

distribution: number | undefined

The distribution number of the the event. - 0: Your organization only - 1: This community only - 2: Connected communities - 3: All communities - 4: Sharing group - 5: Inherit Event

1.8 PillCollection

<div><div><div>C</div></div><div>«Component» PillCollection</div></div>	
Properties	
<div><div><div>o pills: { <code>label?: string undefined</code>; <code>text?: string undefined</code>; <code>icon?: string undefined</code>; <code>class?: string undefined</code>; <code>style?: string undefined</code>; } []</div><div>o class?: string</div></div></div>	


1.8.1 Props

-pills:

```
{ label?: string | undefined; text?: string | undefined; icon?: string | undefined; class?: string | undefined; style?: string | undefined; }[]
```

class: string | undefined

1.9 Boolean

<div>  <div> <div>«Component»</div> <div>Boolean</div> </div> </div>
<p>Displays a boolean value as a text using the <code>{@link Info }</code> component. The background is green if the value is true and red if the value is false.</p>
<div>Properties</div> <p>Displays a boolean value as a text using the <code>{@link Info }</code> component.</p> <ul style="list-style-type: none"> Also parses strings to boolean. String must be either 'true' or 'false'. isTrue?: string boolean Additional classes to be applied to the <code>{@link Info }</code> component. class?: string

Displays a boolean value as a text using the [Info](#) component. The background is green if the value is true and red if the value is false.

1.9.1 Props


isTrue: string | boolean | undefined

Displays a boolean value as a text using the [Info](#) component. Also parses strings to boolean. String must be either 'true' or 'false'.

class: string | undefined

Additional classes to be applied to the [Info](#) component.

1.10 PillNavigation

<div>  <div> <div>«Component»</div> <div>PillNavigation</div> </div> </div>
<div>Properties</div> <ul style="list-style-type: none"> The route that will be displayed in the pill navigation. routes?: Route[] The action that will be displayed in the pill navigation. action: string

1.10.1 Props

routes: Route[] | undefined

The route that will be displayed in the pill navigation.

action: `string`

The action that will be displayed in the pill navigation.

1.11 Button

<div>Ⓒ «Component» Button</div>	
A button with a slot for content.	
Properties	
○	Additional classes to be applied to this component <code>class?: string</code>
Slots	
○	<code>default</code>
Events	
○	<code>click: MouseEvent</code>

A button with a slot for content.

Example:

```
<Button>
  <b>name</b>
  <Icon icon="mdi-light:home" class="w-8 h-8" />
</Button>
```

1.11.1 Props

class: `string` | `undefined`

Additional classes to be applied to this component

1.11.2 Slots

default

1.11.3 Events

click: `MouseEvent`

1.12 Card

<div>Ⓒ «Component» Card</div>	
A card with a slot for content. Sets the default padding and border radius. You can override this by passing your own classes.	
Properties	
○	<code>class?: string</code>
Slots	
○	<code>default</code>

A card with a slot for content. Sets the default padding and border radius. You can override this by passing your own classes.

1.12.1 Props

class: string | undefined

1.12.2 Slots

default

1.13 CardRow

C «Component» CardRow	
If you want too display rows inside of a card, use this component. You should probably just input two children to the slot. Because of the flexbox, the children will at the outsides of the row (justify between).	
o class?: string	Properties
o default	Slots

If you want too display rows inside of a card, use this component. You should probably just input two children to the slot. Because of the flexbox, the children will at the outsides of the row (justify between).

1.13.1 Props

class: string | undefined

1.13.2 Slots

default

1.14 Checkbox

C «Component» Checkbox	
A checkbox component. You should bind to the checked property, to access the value. You could also use the on:change event to listen for changes.	
o Whether the checkbox is checked or not. checked: boolean	Properties
o change: Event	Events

A checkbox component. You should bind to the checked property, to access the value. You could also use the on:change event to listen for changes.

Internal :Uses some tailwind css trickery to make the checkbox value to look like a switch. Basically hides the input and sets the focus state via the label. the div is the actual switch and is moved via the peer-checked class. Where the peer class is set in the input.

1.14.1 Props


checked: `boolean`

Whether the checkbox is checked or not.

1.14.2 Events

change: `Event`

1.15 Select

<div>  «Component» Select </div> <div><code>T extends string;</code></div>
<p>A select component that uses the native select element. The options are passed as a prop and the value is bound to the value prop. The options prop should be a "as const" array of objects with a value and a label property to allow full type safety.</p>
<div>Properties</div> <ul style="list-style-type: none"> ○ The options of the select. The value is the value of the option and the label is the label of the option. <code>options: readonly { value: T; label: string; }[]</code> The value that is currently selected. ○ Because of the template variable you should have full type safety if using consts as options. <code>value: T</code>

A select component that uses the native select element. The options are passed as a prop and the value is bound to the value prop. The options prop should be a "as const" array of objects with a value and a label property to allow full type safety.

1.15.1 Props

`T extends string`


options: `readonly { value: T; label: string; }[]`

The options of the select. The value is the value of the option and the label is the label of the option.

value: `T`

The value that is currently selected. Because of the template variable you should have full type safety if using consts as options.

1.16 Input

<div> «Component»</div> <div>Input</div>
<p>The default input component. You can add a prefix icon inside of the "icon" slot and / or a suffix icon in the "suffix" slot. You should set the name prop, if you want to use this component inside of a form. You can also set the value prop, if you want to set an initial value. Or bind to it if you want to use this outside of a form. You can also set the placeholder prop, if you want to set an placeholder. This is recommended.</p>
<div>Properties</div> <ul style="list-style-type: none"> the placeholder of the input. placeholder: <code>string</code> the name of the input. Used for the label and for form submission name?: <code>string</code> the current value of the input value?: <code>string</code> Additional classes to be applied. class?: <code>string</code>
<div>Slots</div> <ul style="list-style-type: none"> icon suffix
<div>Events</div> <ul style="list-style-type: none"> blur: <code>FocusEvent</code> focus: <code>FocusEvent</code>

The default input component. You can add a prefix icon inside of the "icon" slot and / or a suffix icon in the "suffix" slot. You should set the name prop, if you want to use this component inside of a form. You can also set the value prop, if you want to set an initial value. Or bind to it if you want to use this outside of a form. You can also set the placeholder prop, if you want to set an placeholder. This is recommended.

1.16.1 Props

placeholder: `string`

the placeholder of the input.

name: `string` | `undefined`

the name of the input. Used for the label and for form submission

value: `string` | `undefined`

the current value of the input

class: `string` | `undefined`

Additional classes to be applied.

1.16.2 Slots

icon

suffix

1.16.3 Events

blur: `FocusEvent`

focus: `FocusEvent`

1.17 SideMenuDivider

C	«Component» SideMenuDivider
A divider for the side menu.	
Properties	
o class?:	string

A divider for the side menu.

1.17.1 Props

class: `string` | `undefined`

1.18 SideMenuEntry

C	«Component» SideMenuEntry
The side menu entry component. It can be opened by clicking on it, when the parent side menu is open. If it is open all the children will be displayed as subentries using this component.	
Properties	
o	The name to be displayed in this side menu entry. name: string
o	The icon to be displayed in this side menu entry. icon: string
o	The href to be used in this side menu entry => link to a page. href: string
o	Whether this side menu entry is active or not. => meaning being highlighted. active?: boolean
o	Whether the parent side menu is open or not. isMenuOpen?: boolean
o	The children of this side menu entry. => They will be displayed as subentries. children?: <code>Route[]</code>
o	Whether this side menu entry is a child of another SideMenuEntry. Meaning it being a subentry. isChild?: boolean

The side menu entry component. It can be opened by clicking on it, when the parent side menu is open. If it is open all the children will be displayed as subentries using this component.

1.18.1 Props

name: string

The name to be displayed in this side menu entry.

icon: string

The icon to be displayed in this side menu entry.

href: string

The href to be used in this side menu entry => link to a page.

active: boolean | undefined

Whether this side menu entry is active or not. => meaning being highlighted.

isMenuOpen: boolean | undefined

Whether the parent side menu is open or not.

children: Route[] | undefined

The children of this side menu entry. => They will be displayed as subentries.

isChild: boolean | undefined

Whether this side menu entry is a child of another SideMenuEntry. Meaning it being a subentry.

1.19 SideMenu

 «Component»
SideMenu

The side menu component. It contains the {@link SideMenuEntry } and {@link SideMenuDivider } components. You can override the default SideMenuEntry list display by using the default slot. You can also override the logo by using the "logo" slot. If you do that do not forget the set fade options otherwise it may look weird.

Properties

- The current state of the side menu. It is exportable to be able to change it from other components.
- isOpen?: boolean
- routes?: SideMenuRoute[]
- The current route that is active. Should probably always be the current url (\$page.url.href).
- activeRoute?: string | null

Slots

- logo
- default

The side menu component. It contains the SideMenuEntry and SideMenuDivider components. You can override the default SideMenuEntry list display by using the default slot. You can also override the logo by using the "logo" slot. If you do that do not forget the set fade options otherwise it may look weird.

1.19.1 Props

isOpen: `boolean` | `undefined`

The current state of the side menu. It is exportable to be able to change it from other components.

routes: `SideMenuRoute[]` | `undefined`

The routes to be displayed in the side menu.

activeRoute: `string` | `null` | `undefined`

The current route that is active. Should probably always be the current url (`$page.url.href`).

1.19.2 Slots

logo

default

1.20 TopMenu



«Component»
TopMenu

The top menu component. It contains the `{@link Input}` and `{@link Checkbox}` components. Also the switch for changes between view and edit mode is located here.

Properties

- The mode of the current page. Possible modes are currently "view" and "edit": TODO: maybe extract this to a store?
- `mode?: "view" | "edit"`
- Whether the side menu is open or not. TODO: probably should search for a better solution for this.
- `isOpen?: boolean`

The top menu component. It contains the `Input` and `Checkbox` components. Also the switch for changes between view and edit mode is located here.

1.20.1 Props


mode: `"view"` | `"edit"` | `undefined`

The mode of the current page. Possible modes are currently "view" and "edit": TODO: maybe extract this to a store?

isOpen: `boolean` | `undefined`

Whether the side menu is open or not. TODO: probably should search for a better solution for this.

1.21 Layout

 «Component» Layout	
The basic component for the layout of the application. It contains the <code>{@link SideMenu }</code> , <code>{@link TopMenu }</code> and <code>{@link PillNavigation }</code> components. The body goes inside of the default slot. You can also override the <code>{@link SideMenu }</code> by using the "sideMenu" slot.	
Properties	
<ul style="list-style-type: none"> The routes to be displayed in the side menu. <code>routes: SideMenuRoute[]</code> The current route to be displayed in the <code>{@link PillNavigation }</code>. <code>currentRoute?: Route[]</code> The current action to be displayed in the <code>{@link PillNavigation }</code>. <code>currentAction?: string</code> 	
Slots	
<ul style="list-style-type: none"> <code>sideMenu</code> <code>default</code> 	

The basic component for the layout of the application. It contains the `SideMenu`, `TopMenu` and `PillNavigation` components. The body goes inside of the default slot. You can also override the `SideMenu` by using the "sideMenu" slot.

1.21.1 Props

routes: `SideMenuRoute[]`

The routes to be displayed in the side menu.

currentRoute: `Route[]` | undefined

The current route to be displayed in the `PillNavigation`.

currentAction: `string` | undefined


The current action to be displayed in the `PillNavigation`.

1.21.2 Slots

sideMenu

default

1.22 DatePill

 «Component» DatePill	
Displays a date in a pill. The color of the pill is based on the date. <ul style="list-style-type: none"> - If the date is in the past, the pill will be red. - If the date is over one week in the future, the pill will be green. - If the date is less then one week in the future, the pill will be orange. 	
Properties	
<ul style="list-style-type: none"> The date of the to be displayed. <code>date: Date</code> 	

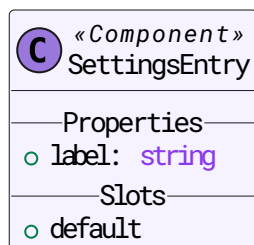
Displays a date in a pill. The color of the pill is based on the date. - If the date is in the past, the pill will be red. - If the date is over one week in the future, the pill will be green. - If the date is less then one week in the future, the pill will be orange.

1.22.1 Props

date: `Date`

The date of the to be displayed.

1.23 SettingsEntry



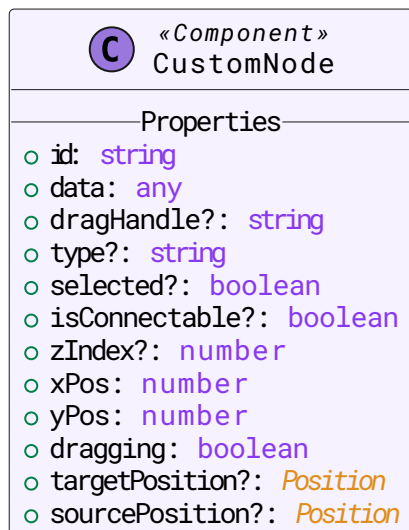
1.23.1 Props

label: `string`

1.23.2 Slots

default

1.24 CustomNode



1.24.1 Props

id: string

data: any

dragHandle: string | undefined

type: string | undefined

selected: boolean | undefined

isConnectable: boolean | undefined

zIndex: number | undefined

xPos: number

yPos: number

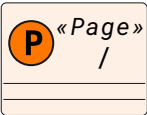
dragging: boolean

targetPosition: Position | undefined

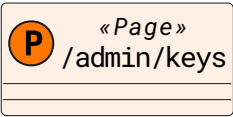
sourcePosition: Position | undefined

2. Pages

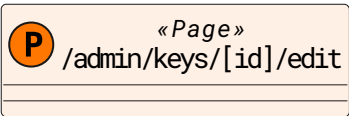
2.1 /



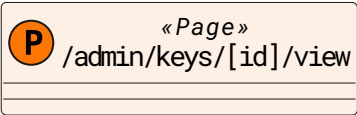
2.2 /admin/keys



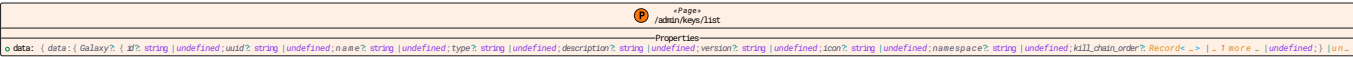
2.3 /admin/keys/[id]/edit



2.4 /admin/keys/[id]/view



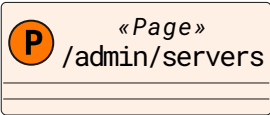
2.5 /admin/keys/list



2.5.1 Props

```
data: { data: { Galaxy?: { id?: string | undefined; uuid?: string | undefined; name?: string | undefined; type?: string | undefined; description?: string | undefined; version?: string | undefined; icon?: string | undefined; namespace?: string | undefined; kill_chain_order?: Record<...> | ... 1 more ... | undefined; } | un...
```

2.6 /admin/servers



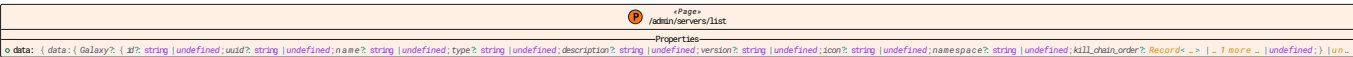
2.7 /admin/servers/[id]/edit



2.8 /admin/servers/[id]/view



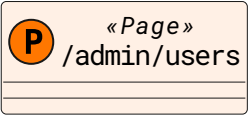
2.9 /admin/servers/list



2.9.1 Props

```
data: { data: { Galaxy?: { id?: string | undefined; uuid?: string | undefined; name?: string | undefined; type?: string | undefined; description?: string | undefined; version?: string | undefined; icon?: string | undefined; namespace?: string | undefined; kill_chain_order?: Record<...> | ... 1 more ... | undefined; } | un...
```

2.10 /admin/users



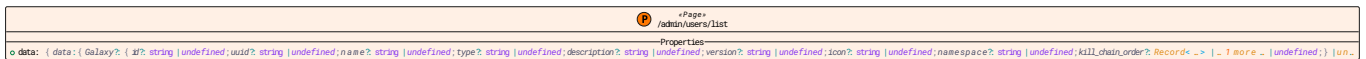
2.11 /admin/users/[id]/edit



2.12 /admin/users/[id]/view



2.13 /admin/users/list



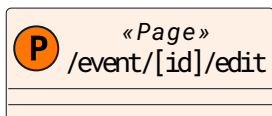
2.13.1 Props

```
data: { data: { Galaxy?: { id?: string | undefined; uuid?: string | undefined; name?: string | undefined; type?: string | undefined;
description?: string | undefined; version?: string | undefined; icon?: string | undefined; namespace?: string | undefined;
kill_chain_order?: Record<...> | ... 1 more ... | undefined; } | un...
```

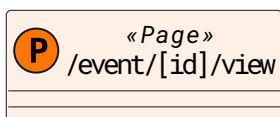
2.14 /event



2.15 /event/[id]/edit



2.16 /event/[id]/view



2.17 /event/list

«Page» /event/list	
Properties	
data:	{ events: ({ id?: string undefined; } & { org_id?: string undefined; distribution?: "0" "1" "2" "3" "4" "5" undefined; info?: string undefined; orgc_id?: string undefined; ... 14 more ...; event_creator_email?: string undefined; } & { ... })[]; header: readonly [...]; tableData: FlatUnion<...>... }

2.17.1 Props

data: { events: ({ id?: string | undefined; } & { org_id?: string | undefined; distribution?: "0" | "1" | "2" | "3" | "4" | "5" | undefined; info?: string | undefined; orgc_id?: string | undefined; ... 14 more ...; event_creator_email?: string | undefined; } & { ...; })[]; header: readonly [...]; tableData: FlatUnion<...>...

2.18 /galaxy

«Page» /galaxy	

2.19 /galaxy/[id]/edit

«Page» /galaxy/[id]/edit	

2.20 /galaxy/[id]/view

«Page» /galaxy/[id]/view	
Properties	
data:	{ galaxy: { Galaxy?: { id?: string undefined; uuid?: string undefined; name?: string undefined; type?: string undefined; description?: string undefined; version?: string undefined; icon?: string undefined; namespace?: string undefined; kill_chain_order?: Record<...> ... 1 more ... undefined; } ... }

2.20.1 Props

data:
 { galaxy: { Galaxy?: { id?: string | undefined; uuid?: string | undefined; name?: string | undefined; type?: string | undefined; description?: string | undefined; version?: string | undefined; icon?: string | undefined; namespace?: string | undefined; kill_chain_order?: Record<...> | ... 1 more ... | undefined; } | ... }

2.21 /galaxy/list

«Page» /galaxy/list	
Properties	
data:	{ galaxies: { Galaxy?: { id?: string undefined; uuid?: string undefined; name?: string undefined; type?: string undefined; description?: string undefined; version?: string undefined; icon?: string undefined; namespace?: string undefined; kill_chain_order?: Record<...> ... 1 more ... undefined; } ... }

2.21.1 Props

```
data: { galaxies: { Galaxy?: { id?: string | undefined; uuid?: string | undefined; name?: string | undefined; type?: string | undefined; description?: string | undefined; version?: string | undefined; icon?: string | undefined; namespace?: string | undefined; kill_chain_order?: Record<...> | ... 1 more ... | undefined; } ...
```

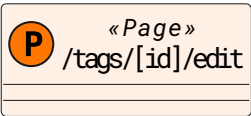
2.22 /settings



2.23 /tags



2.24 /tags/[id]/edit



2.25 /tags/[id]/view



2.26 /tags/list

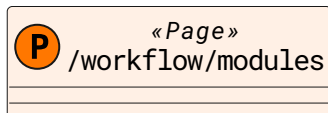
«Page» /tags/list	
Properties	
e data:	{ data: { Tag?: ({ id?: string undefined; } & { name?: string undefined; colour?: string undefined; exportable?: boolean undefined; org_id?: string undefined; user_id?: string undefined; ... 4 more ...; inherited?: number undefined; })[] undefined; }; tableData: FlatUnion<...> > & { header: readonly [...]

2.26.1 Props

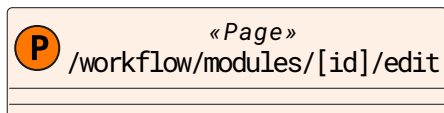
data:

```
{ data: { Tag?: ({ id?: string | undefined; } & { name?: string | undefined; colour?: string | undefined; exportable?: boolean | undefined; org_id?: string | undefined; user_id?: string | undefined; ... 4 more ...; inherited?: number | undefined; })[] | undefined; }; tableData: FlatUnion<...>[]; header: readonly [.....
```

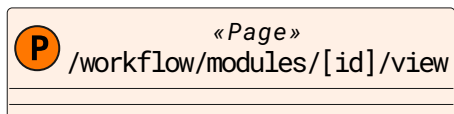
2.27 /workflow/modules



2.28 /workflow/modules/[id]/edit



2.29 /workflow/modules/[id]/view



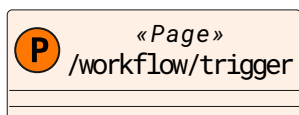
2.30 /workflow/modules/list

«Page»	
/workflow/modules/list	
Properties	
data: { data: Record<string, never> { Galaxy?: { id?: string undefined; uuid?: string undefined; name?: string undefined; type?: string undefined; description?: string undefined; version?: string undefined; icon?: string undefined; namespace?: string undefined; kill_chain_order?: Record<...> ... 1 more...	

2.30.1 Props

```
data: { data: Record<string, never> | { Galaxy?: { id?: string | undefined; uuid?: string | undefined; name?: string | undefined; type?: string | undefined; description?: string | undefined; version?: string | undefined; icon?: string | undefined; namespace?: string | undefined; kill_chain_order?: Record<...> | ... 1 more...
```

2.31 /workflow/trigger



2.32 /workflow/trigger/[id]/edit

P

«Page»

/workflow/trigger/[id]/edit

2.33 /workflow/trigger/[id]/view

P

«Page»

/workflow/trigger/[id]/view

2.34 /workflow/trigger/list

P

«Page»

/workflow/trigger/list

Properties

data: { data: { Galaxy?: { id?: string | undefined; uuid?: string | undefined; name?: string | undefined; type?: string | undefined; description?: string | undefined; version?: string | undefined; icon?: string | undefined; namespace?: string | undefined; kill_chain_order?: Record<...> | ... 1 more ... | undefined; } | un...

2.34.1 Props

data: { data: { Galaxy?: { id?: string | undefined; uuid?: string | undefined; name?: string | undefined; type?: string | undefined; description?: string | undefined; version?: string | undefined; icon?: string | undefined; namespace?: string | undefined; kill_chain_order?: Record<...> | ... 1 more ... | undefined; } | un...

3. Layouts

3.1 /

L «Layout» /
Slots
○ sideMenu
○ default

3.1.1 Slots

sideMenu

default

4. Error Pages

4.1 /

