

Admin Product Picker Component

Admin Product Picker Component

- [How do I use the product picker in my custom module?](#)
- [What are the available props for the product picker?](#)
- [What are the available events for the product picker?](#)

How do I use the product picker in my custom module?

The Product Picker provides a modal that lets users select a single or many products/variants with quantities. Parts of the admin use this such as the Price Lists and Create Order form. Modules such as **aerocargo/upsells** also make use of the Product Picker.

You can use the `<product-picker></product-picker>` Vue component from any admin view inside of your module that extends the official admin layout (`admin::layouts.main`). When using the Product Picker you need to provide a url for the `get-products-url` prop (this value is usually always the `admin.catalog.products.picker` route) and you need to provide a slot for the button that will be used to open the picker.

```
@extends('admin::layouts.main')

@section('content')
    <h2>
        <a href="{{ route('admin.modules') }}" class="btn">@include('admin::icons.back')
Back</a>
        <span class="ml-4">My Module</span>
    </h2>

    <div class="card">
        <product-picker v-slot="picker"
                        get-products-url="{{ route('admin.catalog.products.picker') }}">
            <a href="#" @click.prevent="picker.open" class="btn">Add Product</a>
        </product-picker>
    </div>
@endsection
```

What are the available props for the product picker?

Prop	Type	Description
get-products-url	String	This is the only required prop. This is the URL for the API to get the products from. It's common to set this value to <code>{{ route('admin.catalog.products.picker') }}</code> .
image-url-prefix	String	This is the image factory prefix to use for the product image URLs. This is set by default to use <code>{{ image_factory(60, 60)->contain() }}</code> .
no-image-url	String	This is the image URL to use when a product doesn't have an image. This is set by default to use <code>{{ asset('modules/aerocommerce/admin/no-image.svg') }}</code> .
action-name	String	This is the button text shown on the confirm button of the product picker modal (bottom right). This is set by default to "Add Products".
clear-selected-after-emit	Boolean	
pick-product	Boolean	This is a boolean that defaults to false. When true, products will be allowed to be selected as well as variants.

Prop	Type	Description
product-only	Boolean	This is a boolean that defaults to false. When true, variants will not be listed and not selectable.
pick-only	Boolean	This is a boolean that defaults to false. When true, variants will not need to be in stock to be selectable.
needs-quantity	Boolean	This is a boolean that defaults to true. When false, a quantity will not be able to be set for the selected variants/products.
emit-image	Boolean	This is a boolean that defaults to false. When true, the products/variants image will be emitted when selected.
emit-sku	Boolean	This is a boolean that defaults to false. When true, the products/variants sku will be emitted when selected.
emit-model	Boolean	This is a boolean that defaults to false. When true, the products/variants model will be emitted when selected.
emit-name	Boolean	This is a boolean that defaults to false. When true, the products/variants name will be emitted when selected.
emit-price	Boolean	This is a boolean that defaults to false. When true, the products/variants price will be emitted when selected.

Example

You can make use of the props like you usually would with a [Vue component](#)

```
@extends('admin::layouts.main')

@section('content')
    <h2>
        <a href="{{ route('admin.modules') }}" class="btn">@include('admin::icons.back')
    Back</a>
    <span class="ml-4">My Module</span>
```

</h2>

<div class="card">

<product-picker v-slot="picker"

get-products-url="{{ route('admin.catalog.products.picker') }}"

:max-selection="5"

:pick-product="true"

:emit-name="true"

:emit-model="true"

:emit-sku="true">

Add Product

</product-picker>

</div>

@endsection

What are the available events for the product picker?

Event	Description
products-selected	This event emits when the action button is clicked (bottom right of the modal) and provides an array of all of the selected product(s)/variant(s).
selected	This event emits when the action button is clicked (bottom right of the modal) and is emitted for each product/variant that has been selected - only a single product/variant is provided.

Example

The usual use-case for the Product Picker is to allow users to select products and/or variants to be added to a list. This example code is a Blade view that allows the user to select up to 5 products or variants and then displays them in a list with a custom message. It also allows the user to remove the selected products/variants. The next steps to make this code fully functional would be to wrap the table in a form and to add a submit button that saves the products/variants.

```
@extends('admin::layouts.main')

@section('content')
    <h2>
        <a href="{{ route('admin.modules') }}" class="btn">@include('admin::icons.back')
    Back</a>
        <span class="ml-4">My Module</span>
    </h2>

    <div class="card p-0">
        <table>
            <thead>
                <tr class="header">
```

```

        <th class="whitespace-no-wrap">Model</th>
        <th class="whitespace-no-wrap">Sku</th>
        <th class="whitespace-no-wrap">Message</th>
        <th></th>
    </tr>
</thead>
<tbody>
    <tr v-for="(item, index) in list" :key="item.buyable_id + item.buyable_type" v-
cloak>
        <td class="whitespace-no-wrap">
            <input type="hidden" :name="'items[' + index + '][buyable_id]'"
:value="item.buyable_id">
            <input type="hidden" :name="'items[' + index + '][buyable_type]'"
:value="item.buyable_type">
            <span v-text="item.model"></span>
        </td>
        <td class="whitespace-no-wrap" v-if="item.sku" v-text="item.sku"></td>
        <td class="whitespace-no-wrap" v-else><span class="text-grey">—</span></td>
        <td class="whitespace-no-wrap">
            <input type="text" :name="'items[' + index + '][message]'" v-
model="item.message" placeholder="Message">
        </td>
        <td>
            <div class="flex items-center justify-end">
                <a href="#" @click.prevent="list.splice(index,
1)">@include('admin::icons.bin')</a>
            </div>
        </td>
    </tr>
    <tr v-if="list.length === 0">
        <td colspan="4">No data to show</td>
    </tr>
</tbody>
</table>

<product-picker v-slot="picker"
    v-if="list.length < 5"
    get-products-url="{ route('admin.catalog.products.picker') }"
    :max-selection="5"
    :pick-product="true"
    :emit-name="true"

```

```

        :emit-model="true"
        :emit-sku="true"
        @selected="selected">
        <a href="#" @click.prevent="picker.open" class="btn m-4">Add Product</a>
    </product-picker>
</div>
@endsection

@push('scripts')
<script>
    window.AeroAdmin.addData('list', []);

    window.AeroAdmin.addMethod('selected', function (item) {
        if (this.findIndex(item.buyable_id, item.buyable_type) === -1) {
            this.list.push({
                ...item,
                message: '',
            });
        } else {
            this.$notify({
                type: 'error',
                title: (item.sku ?? item.name) + ' has already been added to the list',
            });
        }
    });

    window.AeroAdmin.addMethod('findIndex', function (id, type) {
        return this.list.findIndex((l) => l.buyable_id === id && l.buyable_type === type);
    });
</script>
@endpush

```