

Step 1 - Display content of a Ride

Tutorial path

We will customize the Google map later, now let's inject and display all Points of interest within the view.

- 1 [Points Of Interest](#)
 - 1.1 [Create the Point Of Interest \(POI\) Content Type](#)
 - 1.2 [Edit the Ride Content Type](#)
- 2 [Ride view](#)
 - 2.1 [Display the list of POI](#)
 - 2.1.1 [Add a new parameter to your override rule](#)
 - 2.2 [Create your Point of interest line view](#)

Points Of Interest

Go to Admin Panel > Content Types, and under the "Content" group, create the Point Of Interest Content Type.

Create the Point Of Interest (POI) Content Type

A geographical location rides go through. Each ride may be related to as many points of interest as needed.

Name: Point of interest
Identifier: point_of_interest
Content name pattern: <name>

Then create all fields with the following information:

- **Name:** identifier **name**; field type **textLine** (**Required** / **Searchable** / **Translatable**)
- **Description:** identifier **description**; field type **Rich Text** (**Required** / **Searchable** / **Translatable**)
- **Photo:** identifier **image**; field type **Image** (**Required** / **Searchable**)
- **Legend:** identifier **legend**; field type **Rich Text** (**Required** / **Searchable** / **Translatable**)
- **Place:** identifier **place**; field type **MapLocation** (**Required** / **Searchable**)

The **content name pattern** defines how the name and URL part of Content items of this type will be built. It may include static strings, as well as references to field definitions, using their identifier.

The value of the fields referenced in the pattern will be used to build the name. Most Field Types are able to render a textual representation of their value, but be aware that it is not implemented for some of them ([Selection FieldType](#), [Relation FieldType](#), [RelationList FieldType](#)).

Create some Points Of Interest in the Content tree. Note that you will need pictures (for the Photo, the image Field) to represent these Points Of Interest.

Edit the Ride Content Type

Now, validate and edit the Ride in order to add a Relation (multiple) between the two Content Types.

☐ Points of Interest [ezobjectrelationlist] (id: 194)

Position

Name

Identifier

Description

Required ☐

Searchable ☒

Translatable ☒

Field group

Default location

Allowed content types

Then link some Points Of Interests to a Ride in the Admin interface.

Ride view

Display the list of POI

By default, there are only 4 variables in a view: noLayout, viewbaseLayout, content and location.

It is possible to inject whatever variable you want in a specific view.

You will find more info here: [How to use a custom controller to display a content item or location](#) and [View provider configuration](#).

Add a new parameter to your override rule

We have already set an override rule for ContentType Ride and we will need to modify it to use our own action injecting the list of Point of interest content.

```

ezplatform.yml
default:
  content_view:
    full:
      full_ride:
        template:
          "content/view/full/ride.html.twig"
        match:
          Identifier\ContentType: "ride"

```

Create your Point of interest line view

Now, we need to create the line view for Point of interest.

Declare a new override rule into your app/config/ezplatform.yml:

ezplatform.yml

```
system:
  site_group:
    default:
      content_view:
        line:
          line_point_of_interest:
            template:
              'content/view/line/point_of_interest.html.twig'
            match:
              Identifier\ContentType:
                ['point_of_interest']
```

Create your template for the line view of a Point of interest app/Resources/views/line/point_of_interest.html.twig:

point_of_interest.html.twig

```
<div class="col-xs-4 photos-box">
    <a href="#bikeModal{{ content.id }}"
class="portfolio-link" data-toggle="modal">
        <div class="caption">
            <div class="caption-content">
                <i class="fa fa-search-plus fa-3x"></i>
            </div>
        </div>
        {{ ez_render_field( content, 'image', {
parameters: { 'alias': 'small'},
                                attr: {
'class': 'img-responsive img-rounded' }}} ) }}
        {##}
    </a>
</div>

{# MODAL #}
<div class="bike-modal modal fade" id="bikeModal{{
content.id }}" tabindex="-1" role="dialog"
aria-hidden="true">
    <div class="modal-content">
        <div class="close-modal" data-dismiss="modal">
            <div class="lr">
                <div class="rl">
                    </div>
                </div>
            </div>
            <div class="container">
                <div class="row">
                    <div class="col-xs-8 col-xs-offset-2">
                        <div class="modal-body text-center">
                            <h2>Photo: {{ ez_content_name(
content ) }}</h2>
                            <hr class="featurette-divider">
                            {{ ez_render_field( content,
'image', { parameters: { 'alias': 'large'},
                                attr: { 'class': 'img-responsive img-rounded' }}} ) }}
                            {##}
                            {{ ez_render_field( content,
'description', { attr: { 'class': 'padding-box
text-justify' }}} ) }}
                            {#<p class="padding-box
text-justify"></p>#}
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
```

Previous: [Working on the Ride introduction page](#)

Next: [Step 2 - Display the list of Rides on the homepage](#)