

# LeBlender

## *Documentation*



v 0.9.x

## Presentation

**LeBlender** is a property editor that helps you manage and create Umbraco Grid editors in a super easy and fast way without any lines of codes.

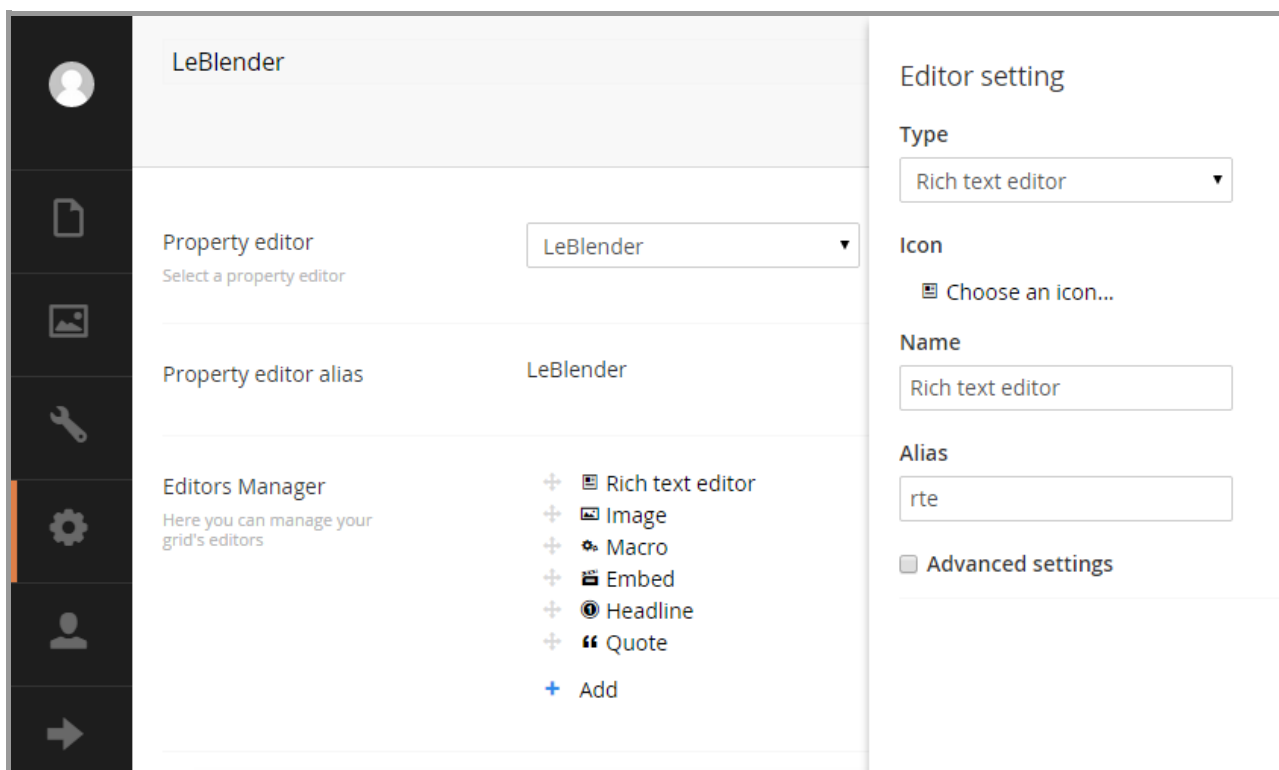
It comes with several features:

- **LeBlender Editors Manager:** UI for Grid editors management.
- **LeBlender Editor:** Super powerful editor for advanced editor creation on the fly.

## LeBlender Editors Manager

Basically, **LeBlender Editor Manager** is a simple UI to manage editors' configurations. Through it, you can create, update, sort and remove editors for Umbraco Grid.

Every time you change a configuration through the UI, the grid's configuration file `/config/grid.editors.config.js` will be updated automatically. You don't have to write any JSON.

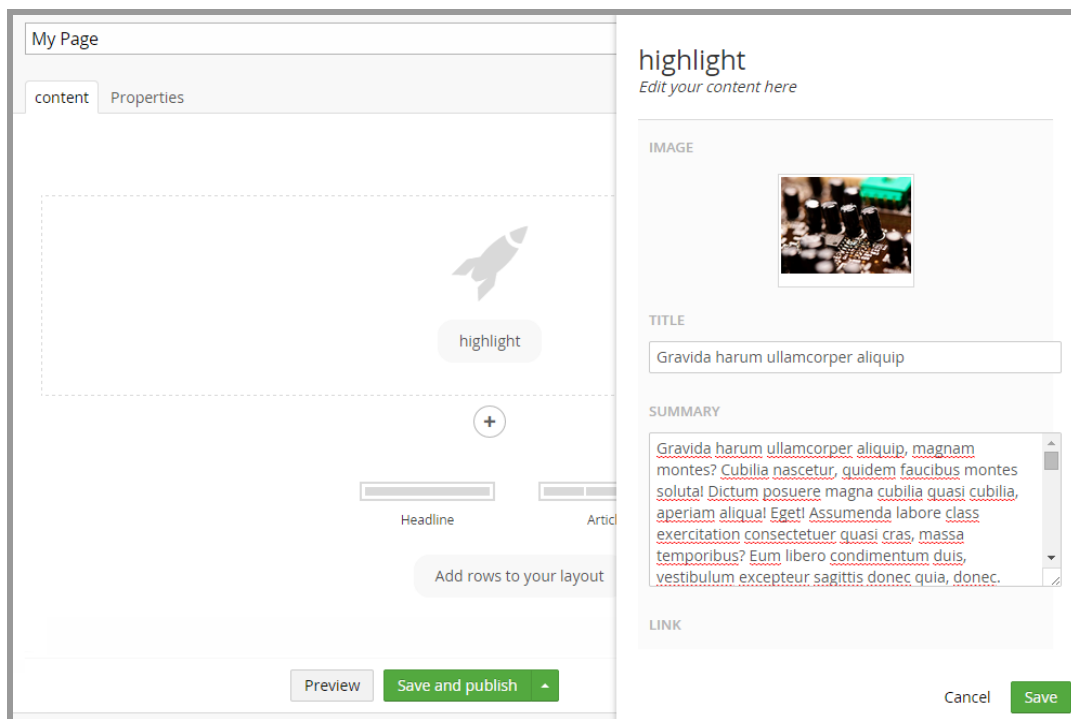
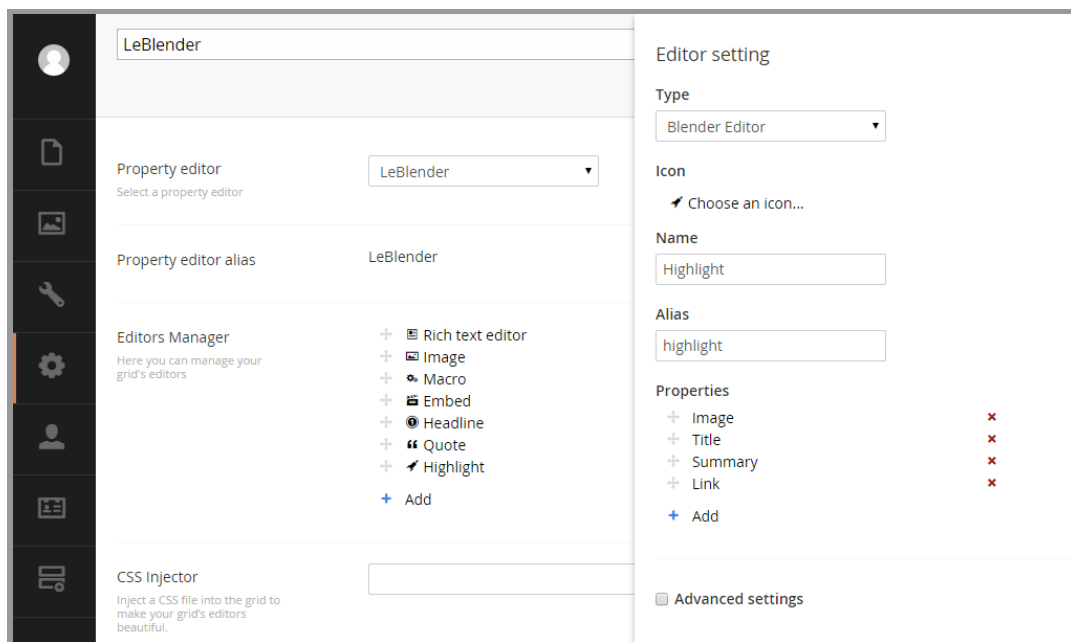


### Important note

Sometimes, you will need to refresh your browser cache to make the Grid take into account your new configurations.

## LeBlender Editor

**LeBlender Editor** is an additional editor that allows you to create Grid editors on the fly with specific sets of properties.



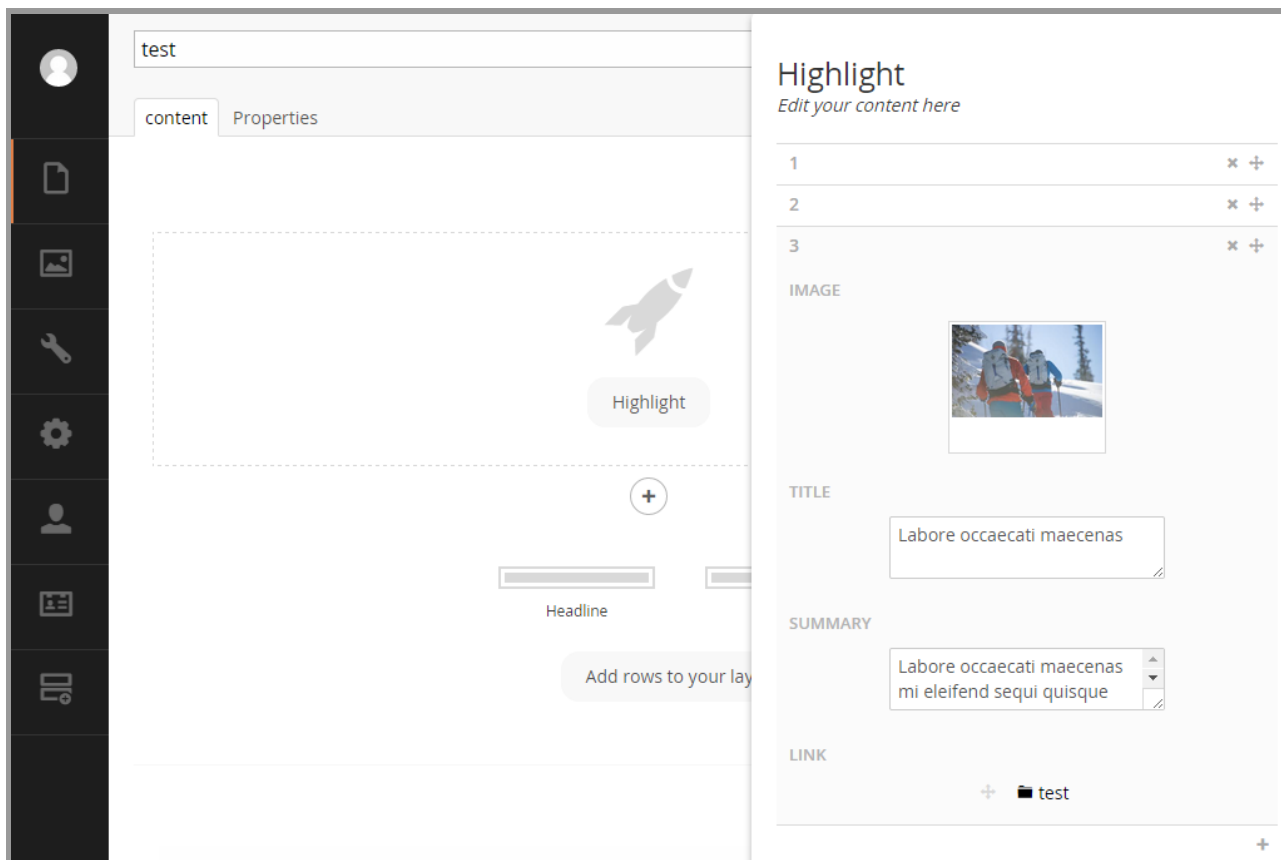
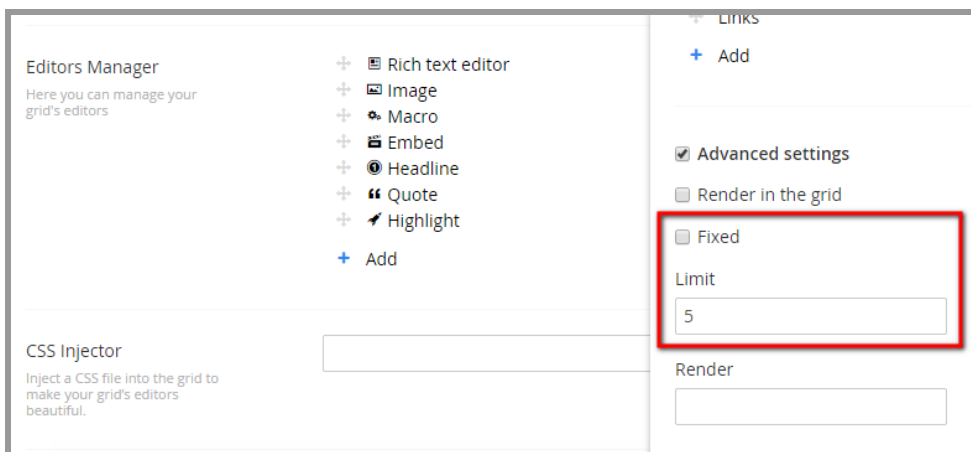
By default, properties can be: **textstring, textarea, boolean, mediapicker, multi mediapicker, content picker.**

It is, of course, super easy to extend with your own property.

You can also configure **LeBlender Editor** to return list of items, this way you can easily create editors for **sliders, carousels, collapsed lists, tab lists** and others.

In this case, you just have to go to **LeBlender Editor's** advance settings and configure:

- Fixed: Whether the number of items is **fixed**, i.e. exactly the number specified in the Limit field, or **variable**, i.e. any number between zero and the number specified in the Limit field.
- Limit: **The maximum number of items** for the list.



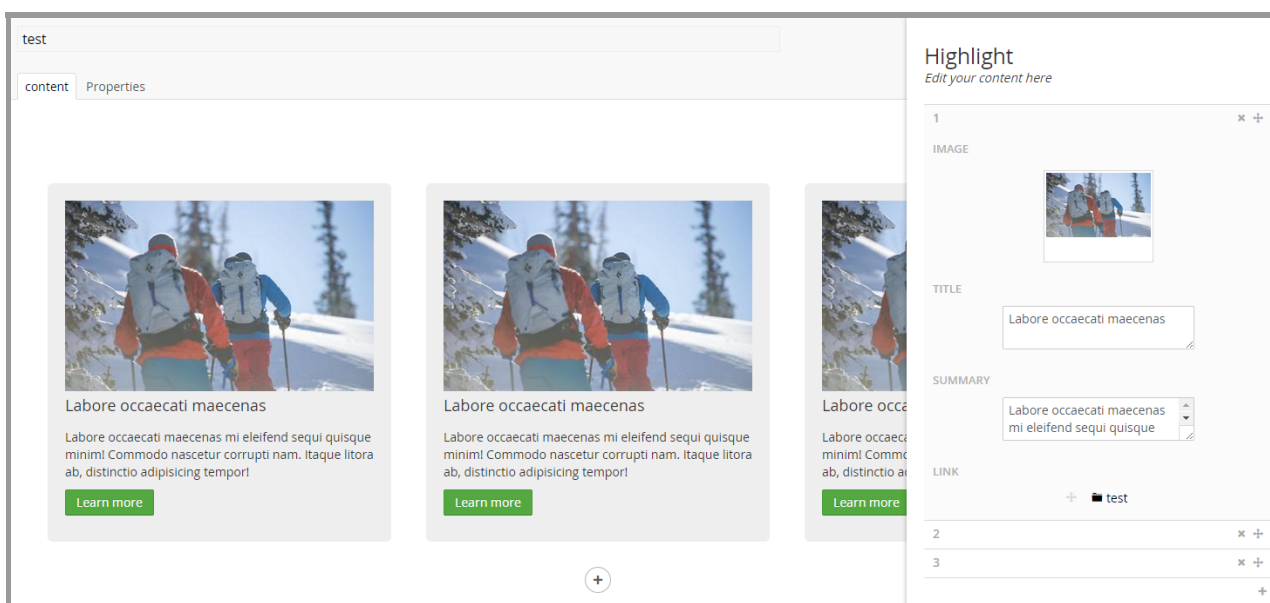
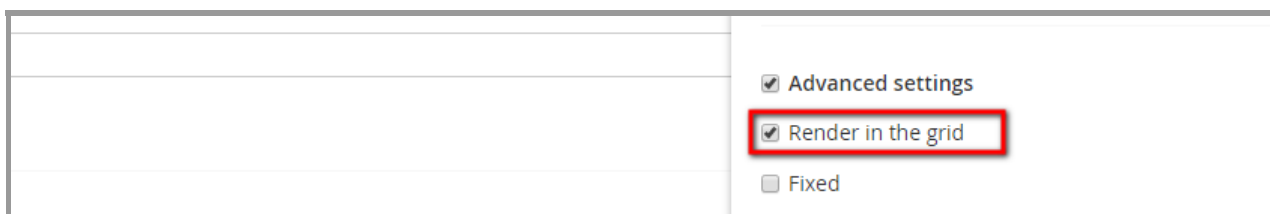
On the frontend side, by default the **LeBlender** engine looks in the folder **/views/partials/grid/editors** for a partial view with the same name as the editor. You can of course define your own custom path instead, through the **Render** field in the advanced settings of the **LeBlender Editor**.

The partial view will receive a strongly typed object of type **BlenderModel** that you can use to easily access the editor's data. You can have a look on some samples in: */App\_Plugins/Lecoati.LeBlender/views\_samples*

```
@inherits UmbracoViewPage<Lecoati.LeBlender.Extension.Models.BlenderModel>

@foreach (var item in Model.Items)
{
    <div>
        
        <h3>@item.GetValue("title")</h3>
        <p>@item.GetValue("summary")</p>
        <p>
            <a href="@Umbraco.TypedContent(item.GetValue<string>("link")).Url">
                Learn more
            </a>
        </p>
    </div>
}
```

The **LeBlender** Editor also offers you the option to render the partial view on the back-end in the Grid:



## Other Useful use

Another use for **LeBlender Editor** is as a wrapper for Partial View. This way, you can easily use and render your partial views directly into the grid.

You can optionally use the properties of **LeBlender Editor** as parameters for your Partial View.

## How to install

### How to install

1. Download LeBender Package  
<https://our.umbraco.org/projects/backoffice-extensions/leblender>
2. Install it through the Umbraco Backend
3. Create a new Datatype of type LeBlender

## We are working on

- Custom controllers for **LeBlender Editors**
- Optional cache for **LeBlender Editors**
- AngularJS services to customize the grid on the back-end depending on the cell/row settings