

Bootstrap PHP/ jQuery Layout Editor

Table of Contents:

[Table of Contents:](#)

[Foreword:](#)

[Purpose:](#)

[Nearest Existing Examples:](#)

[Example 1 - Layoutit](#)

[Example 2 - WordPress Layout Editor](#)

[Widgets](#)

[Icon Picker](#)

[Saved JSON Example](#)

[Technologies required:](#)

Foreword:

I have created this specification document to outline the job specification so that you have a far better idea of the product specifications.

You will be working with the client who has a firm understanding of PHP & jQuery to create this project, however is lacking the time (*and current ability if he's honest*) to create this project by himself.

Your role is to create the layout editor, jQuery logic and the PHP AJAX controllers.

A "rough" wireframe using Bootstrap 3 will be provided as a base, however will need you to add in the magic to make it all work and styling will be required.

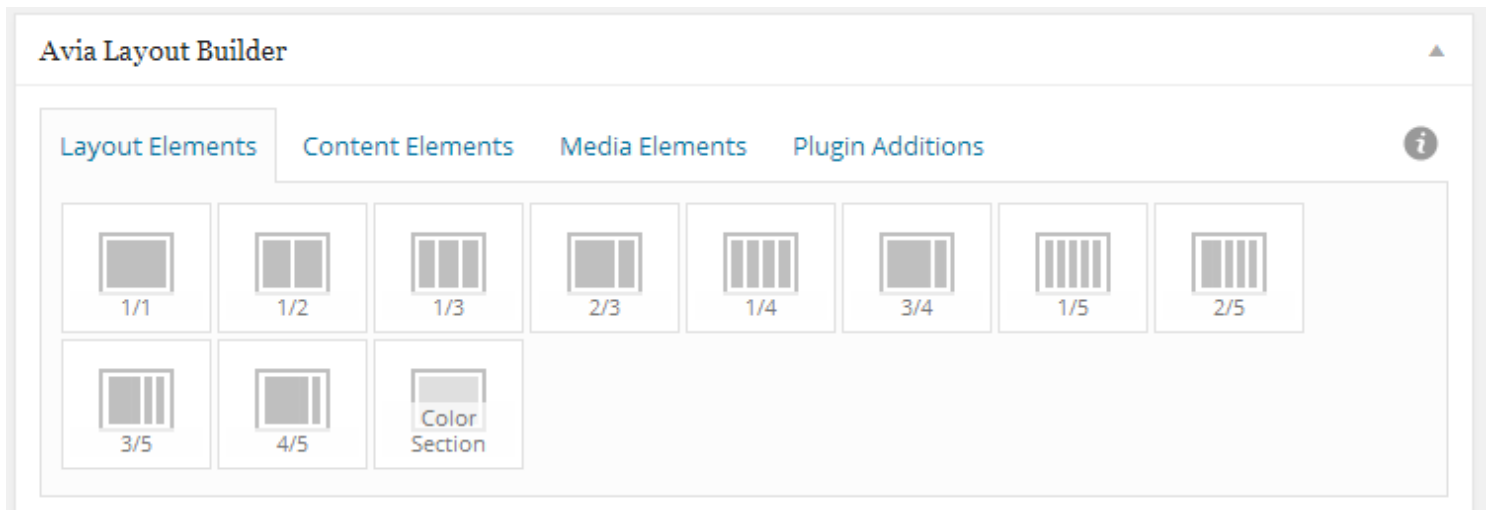
Purpose:

To build layout editor using drag and drop widgets from a menu bar and then save/update the layout as a json array into MySQL table.

Bootstrap 3 will be used and each widget will have sample data included and may include options that need to be shown in a modal window for editing.

A preview option will be available which hides the menu bar and any buttons or additional layout tools.

Existing Example:



The top or side menu will have 4 tabbed sections included:

1. **Layout**

This section includes the row formats, eg 12, 6/6, 8/4, etc... that can be dragged to the working area

2. **Widgets**

Numerous widgets (*each already specified with additional parameters, see preceding section*) which can be dragged and dropped into columns, moved around using a drag option and also removable from the layout.

Note: Some widgets can only be added once to a layout

3. **Design**

This section will include numerous colour pickers using the jQuery plugin called "Spectrum Colour Picker" that update the preview option.

Note: Most of the jQuery required for this has already been written

4. **Settings**

This section will have additional settings that have no effect on the layout or preview panels

Along with a save button (*saves layouts as JSON, see later section*) and a preview mode which removes the widget menu bar and shows the layout with no layout comments or markup and populates with example information (*which will be provided*).

Nearest Existing Examples:

Example 1 - Layoutit

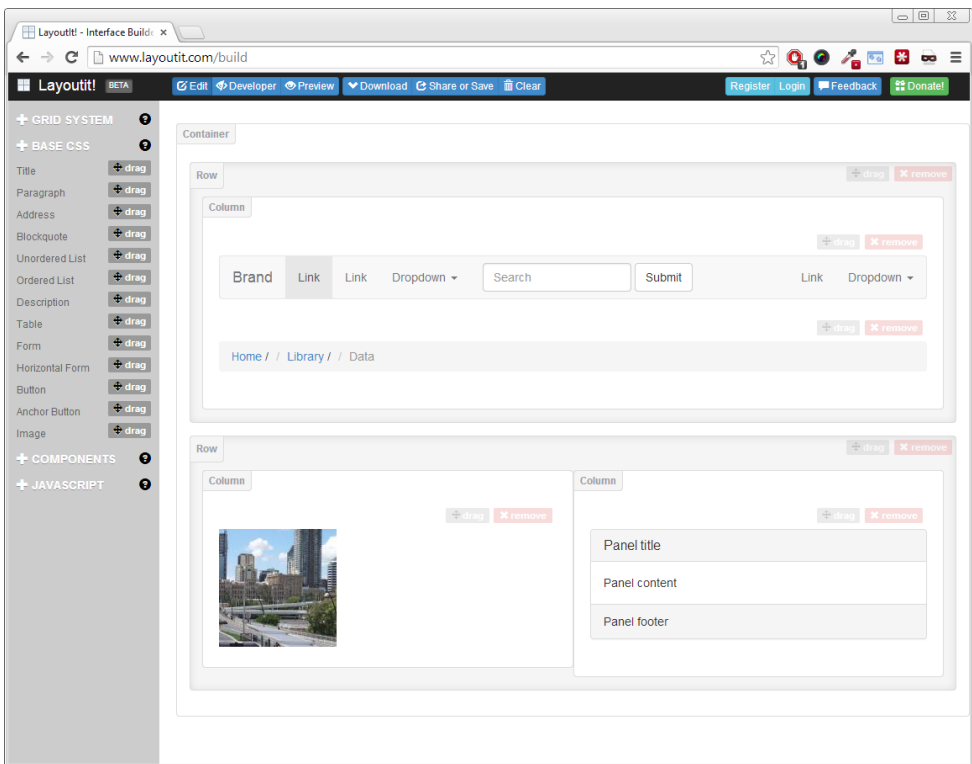
The site at <http://www.layoutit.com/build> is the nearest I can find to the features I'm looking for.

In the screenshot below, you can drag and drop elements from a sidebar menu.

So for example a 12 with row, add in a nav bar, breadcrumbs, an image and panels.

Note:

Such draggable elements will be clearly defined and are referred to as “widgets“



Example 2 - WordPress Layout Editor

There is a layout editor for a Wordpress theme that has similar functionality (*source code can be provided*) that has a very similar style of what I'm looking for.

The theme is called enfold ([see here](#)) and in the screenshots on the following page, you'll notice that:

- There are 4 tabs
- Each tab has options within them
- These options can be dragged to the content area
- The column layouts would always = 12
- On the content elements tab, this would be the “widgets”

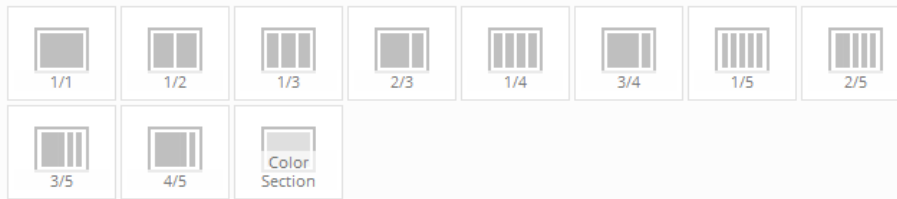
Avia Layout Builder

Layout Elements

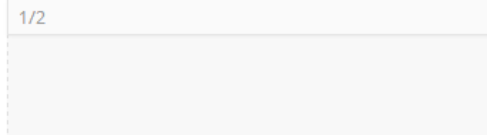
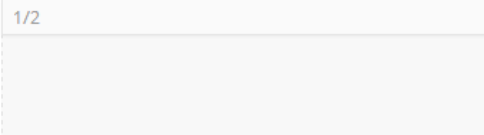
Content Elements

Media Elements

Plugin Additions



Templates



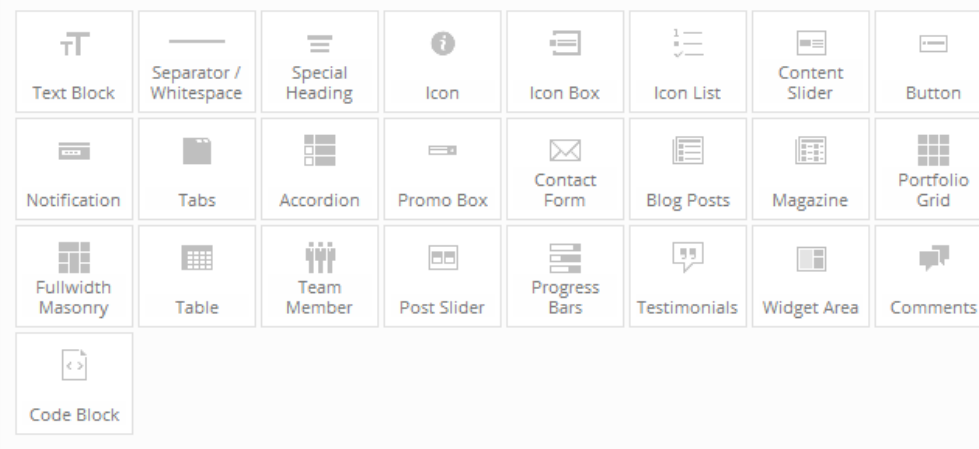
Avia Layout Builder

Layout Elements

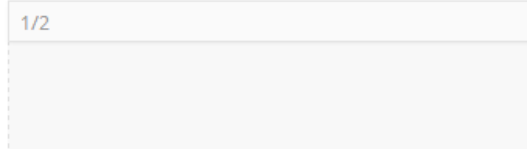
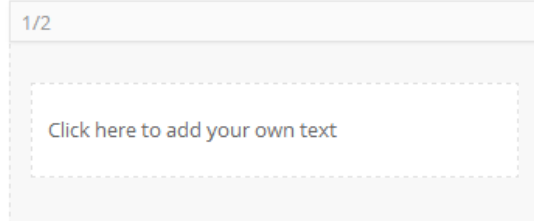
Content Elements

Media Elements

Plugin Additions



Templates



Widgets

Each widget will have predefined values and if viewing a saved layout, must be loaded and displayed for the user that is viewing the layout editor.

There are 15 widgets to be added and the PHP framework built **must allow** the addition of future widgets to be added easily.

Each widget has clearly defined settings and **have already been specified** (*so should be very little “vagueness” here!*) and details will be provided on your successful application.

Any configurable options for a widget should be editable using a button called settings next to the widget, which opens a modal window to set the options that widget. These settings should be saved with the layout.

A widget may have:

- Extra settings that can be edited in a modal window
- Text or textarea boxes for the user to enter input text
- Hidden values (non user editable)
- Only be allowed to added once to a layout (*with appropriate user message provided or disabling of the widget in the menu if already in the layout builder*).

Icon Picker

Some of the widgets require an icon to be selected.

The icons from bootstrap here <http://getbootstrap.com/components/#glyphicons> would ideally appear in a pop up window/layer where the user can select the icon to be used.

If possible the addition of selected Font Awesome icons would be desirable (<http://fontawesome.github.io/Font-Awesome/icons/>)

Saved JSON Example

Below is an example of the formatting of the saved json array to be saved into the database.

From the json you'll see that this layout has:

- 2 rows
- row-1 has the 6 by 6 layout
- row-1 has one widget in each section (content-1)
- widget options are stored within each “content” block
- row-2 has two columns 12 wide
- row-2 first column has 1 widget
- row-2 second column has 1 widget

```
{  
  "id" : "1",  
  "userid" : "1",
```

```

"layout" : {

    "row-1" : {

        "columns" : "6-6",
        "column-1" : {
            "width" : "6",
            "content-1" : {

                "widget-type" : "logo",
                "widget-options" : {

                    "option1" : "options here",
                    "option2" : "options here",
                    "url" : "http://somesite.com/logo.png"

                }

            }
        },
        "column-2" : {
            "width" : "6",
            "content-1" : {

                "widget-type" : "search",
                "widget-options" : {

                    "type" : "store",
                    "option1" : "options here",
                    "option2" : "options here",

                }

            }
        }
    },
    "row-2" : {

        "columns" : "12",
        "column-1" : {
            "width" : "12",
            "content-1" : {

                "widget-type" : "navigation",
                "widget-options" : {

                    "menu_name" : "Navigation Menu",
                    "option1" : "options here",
                    "option2" : "options here"

                }

            }
        },
        "column-2" : {
            "width" : "12",
            "content-1" : {

```

```
        "widget-type" : "breadcrumbs",
        "widget-options" : {

            "option1" : "options here",
            "option2" : "options here"

        }
    }
}
}
```

Technologies required:

- jQuery
- Bootstrap
- JQuery UI
- MySQL (*for simplicity, the table will have 3 columns id, userid & settings*)
- Spectrum Colour Picker (<http://bgrins.github.io/spectrum/>)
- AJAX controllers to be written in PHP
- Pure PHP is desired. If a PHP framework is to be used, this must **clearly be stated** in your application

Include this in your application for my immediate attention!

When applying for this job, please include the phrase “**I’ve read the pre-specification**” this will help you and myself identify who has actually read this document and who has not.