

WORDCAMP
LISBOA 2023

Blockworld

Survival Guide for a PHP developer

Arūnas Liuiza

“Learn JavaScript, deeply.”



- Matt Mullenweg, State of the Word 2015



“Damn, that’s HARD..”



- a backend developer

How to survive in Gutenberg world as a PHP developer?



Option 1

Give Up



WORDCAMP
LISBOA 2023

Just learn the damn JavaScript

It's not **that** hard...

[@wordpress/create-block](https://github.com/WordPress/create-block)

Officially supported tool for scaffolding a WordPress plugin that registers a block.

It generates PHP, JS, CSS code, and everything you need to start the project. It also integrates a modern build setup with no configuration.

<https://github.com/youknowriad/blocky>

```
$ npx @wordpress/create-block todo-list
```

```
$ cd todo-list
```

```
$ npm start
```



Option 2

Collect



WORDCAMP
LISBOA 2023

Use block collection plugins

There is a **collection** of them out there, already.

- [Genesis Blocks](#) (nee [Atomic Blocks](#))
- [CoBlocks](#)
- [GhostKit](#)
- [ACF Blocks](#)
- [GetWid](#)
- [Kadence Blocks](#)
- [Ultimate Blocks](#)
- [Stackable](#)

And many many more...



Option 3

Use ACF, Luke...



WORDCAMP
LISBOA 2023

No, really.

Well, a block **builder** plugin. There is a bunch of those as well.

- [Advanced Custom Fields](#) (**PRO**)
- [Genesis Custom Blocks](#)
(*nee* [Block Lab](#))
- [Lazy Blocks](#)

And others...



Advanced Custom Fields

Pros

- Big user base
- Most field types

Cons

- Most code involved
- Only in PRO version

The screenshot displays the ACF 'Edit Field Group' interface for a 'Testimonial' block. The top navigation bar includes 'ACF', 'Field Groups', 'Tools', and 'Updates'. The main content area is divided into two sections: 'Fields' and 'Settings'.

The 'Fields' section contains a table with the following data:

#	Label	Name	Key	Type
1	Testimonial	testimonial	field_63203a723a62b	Text Area
2	Author	author	field_63203b3eae7dc	Text
3	Role	role	field_63203b47ae7dd	Text
4	Image	image	field_63203b4cae7de	Image
5	Color Settings		field_63203b76ae7df	Accordion
6	Background Color	background_color	field_63203b83ae7e0	Color Picker
7	Text Color	text_color	field_63203b99ae7e1	Color Picker

The 'Settings' section is currently on the 'Location Rules' tab. It shows a rule configuration under the heading 'Rules' and 'Show this field group if'. The rule is: 'Block is equal to Testimonial'.



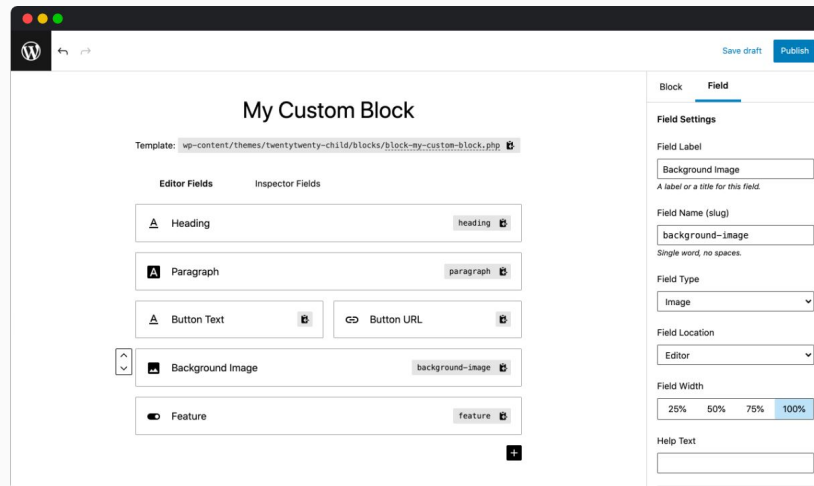
Genesis Custom Blocks

Pros

- Gutenberg specific

Cons

- Has a robust free version
- Block import/export - only in PRO



Lazy Blocks

Pros

- Gutenberg - specific
- Full control via UI
- Full control via code
- Most field types in free version

Cons

- Conditional logic only in PRO

The screenshot displays the Gutenberg editor interface for an 'Example Block'. The main area shows a preview of the block with three content controls: 'Image', 'Button Label', and 'Button URL'. Below these is a 'Code' editor showing PHP conditional logic for rendering the block's content based on the presence of attributes.

```
1 <?php if ( !isset( $attributes['image'] ) ) : ?>
2     <img src="" alt="" />
3     <img src="" alt="" />
4 </?php
5
6 <?php if ( !isset( $attributes['button-label'] ) ) : ?>
7     <button type="button" class="button button-primary">
8         <?php echo esc_html( $attributes['button-label'] ) ?>
9         <?php echo esc_html( $attributes['button-label'] ) ?>
10    </button>
11 </?php
12 <?php else : ?>
13 <?php else : ?>
14     <?php echo esc_html( $attributes['button-label'] ) ?>
15 </?php else : ?>
```

The right-hand side of the image shows the 'Block Control' sidebar. It includes fields for 'Slug' (example-block), 'Icon', 'Category' (Design), 'Keywords' (example, block), and a 'Description' field containing the text: 'Example block that helps you to get started with Lazy Blocks plugin'. There are also expandable sections for 'Supports' and 'Condition'.



Option 4

PHP!



WORDCAMP
LISBOA 2023

Yes, we can!

Not an option directly in WordPress Core, but we can use some tools.



WORDCAMP
LISBOA 2023

GutenFactory

A piece of history.

Built during Contributors Day at
WordCamp Norrköping 2018

An early attempt on purely
PHP-based Gutenberg block
creator

```
add_filter( 'gutenfactory_blocks', 'my_awesome_block' );
function my_awesome_block( $blocks ) {
    $blocks['arunas/my_awesome_block'] = [
        'name'          => 'My Awesome Block',
        'category'      => 'common',
        'style'         => plugins_url( 'my-awesome-block.css', __FILE__ ),
        'editor_style' => plugins_url( 'my-awesome-block-editor.css', __FILE__ ),
        'callback'     => 'my_awesome_block_render',
        'fields'       => [
            // define the fields here.
        ]
    ];
    return $blocks;
}
```



Advanced Custom Fields

PRO version, 5.8 or higher

[acf_register_block_type\(\)](#)

Fields have to be built via UI, but can then be loaded via JSON files

```
add_action('acf/init', 'my_register_blocks');
function my_register_blocks() {
    // check function exists.
    if( function_exists('acf_register_block_type') ) {
        // Register a testimonial block.
        acf_register_block_type(array(
            'name'           => 'testimonial',
            'title'          => __('Testimonial'),
            'description'    => __('A custom testimonial block.'),
            'render_template' => 'template-parts/blocks/testimonial/testimonial.php',
            'category'       => 'formatting',
        ));
    }
}
```



ACF Builder

A PHP Library to easily build ACF fields via code

<https://github.com/StoutLogic/acf-builder>



```
$banner = new StoutLogic\AcfBuilder\FieldsBuilder('banner');  
$banner  
    ->addText('title')  
    ->addWysiwyg('content')  
    ->addImage('background_image')  
    ->setLocation('block', '==', 'acf/testimonial')  
        ->or('block', '==', 'acf/anotherone');  
  
add_action('acf/init', function() use ($banner) {  
    acf_add_local_field_group($banner->build());  
});
```



Lazy Blocks

a Gutenberg blocks visual constructor

The ACF of Gutenberg age

UI based, but has an option to export a PHP snippet

```
if ( function_exists( 'lazyblocks' ) ) :  
    lazyblocks()->add_block( array(  
        'title' => 'Custom block',  
        'slug' => 'lazyblock/custom-block',  
        'supports' => array(  
            'html' => false,  
            'multiple' => true,  
        ),  
        'controls' => array(  
            // define fields  
        ),  
        'code' => array(  
            // editor/frontend templates or callbacks, assets  
        ),  
    ) );  
endif;
```



Lazy Blocks Builder

PHP library to simplify block
declarations

<https://github.com/ideag/lazy-blocks-builder>

```
if ( function_exists( 'lazyblocks' ) ) :  
  
    $block = new Lazy_Blocks_Builder( 'Custom Block' );  
    $block  
        ->setOption( 'description', 'A sample block for WordCamp' );  
    $block  
        ->addText( 'heading' )  
            ->setDefault( 'Big heading here' )  
            ->setRequired()  
        ->addRichTextEditor( 'copy' )  
            ->setDefault( 'Some lorem ipsum here as a placeholder' )  
    ;  
    $block->setSingleCallback( function( $data ) { /* ... */ } );  
    $block->build();  
  
endif;
```



Missing pieces



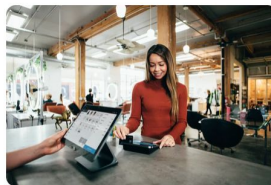
Tailwind CSS

A utility-first CSS framework

Define everything via CSS classes

Works **well** without any build process

Works **amazingly** if you do add a build step



MARKETING

Finding customers for your new business

Getting a new business off the ground is a lot of hard work. Here are five ideas you can use to find your first customers.

```
<div class="md:flex">
  <div class="md:flex-shrink-0">
    
  <div class="mt-4 md:mt-0 md:ml-6">
    <div class="uppercase tracking-wide text-sm text-indigo-600 font-bold">Mar
    <a href="#" class="block mt-1 text-lg leading-tight font-semibold text-gra
    <p class="mt-2 text-gray-600">Getting a new business off the ground is a l
    </div>
  </div>
</div>
```



Alpine.js

A tiny JS declarative and reactive JS framework

You get to keep your DOM, and sprinkle in behaviour as you see fit.

Can be loaded from CDN, without building or bundling.

```
<div x-data="{ open: false }">
  <button @click="open = true">
    Open Dropdown
  </button>
  <ul
    x-show="open"
    @click.away="open =
false"
    >
    Dropdown Body
  </ul>
</div>
```



Thanks!
Any questions?

Arūnas Liuiza

@arunaswp

ask@arunas.co

