

Corpress - Documentation

Thank you for purchasing this theme. If you have any questions that are beyond the scope of this help file, please feel free to email via the user page contact form on this page. Thanks so much!

Scope of free support

We can help you with:

- Responding to questions or problems regarding the item and its features
- Fixing any discovered item's bugs
- Providing updates to ensure compatibility with new software versions

HTML Structure

This theme is built upon bootstrap framework and is fully responsive. It adapts to window resolution and looks good on all devices including iPad and iPhone.

See for yourself (resize your browser).

If you would like to read more about bootstrap framework click here :

[Bootstrap Documentation](#)

If you would like to edit the color, font, or style of any elements, you would do the following:

```
1. #primaryContent a {color: #someColor;}
```

If you find that your new style is not overriding, it is most likely because of a specificity problem. Scroll down in your CSS file and make sure that there isn't a similar style that has more weight.

I.E.

```
1. #wrap #primaryContent a {color: #someColor;}
```

So, to ensure that your new styles are applied, make sure that they carry enough "weight" and that there isn't a style lower in the CSS file that is being applied after yours.

CSS Files and Structure

Colors

The theme includes a css file wich you can use to edit the colors.

You just need to add the corresponding CSS file to the head of the page like this:

```
1. <head>
2.   (..)
3.   <link rel="stylesheet" type="text/css" href="css/red.css">
4.   (..)
5. </head>
```

Main Menu structure

Just change brand name, upload your logo to images/content/ folder and you're ready to go!

```
1. <nav class="navbar yamm" role="navigation">
2.   <div class="container">
3.     <div class="navbar-header hidden-xs">
4.       <a class="navbar-brand" href="index.html"></a>
6.     </div>
7.     <div class="collapse navbar-collapse no-transition" id="bs-
8. example-navbar-collapse-1">
9.       <a href="#" class="showHeaderSearch hidden-xs"><i class="fa
10. fa-fw fa-search"></i></a>
11.     <ul class="nav navbar-nav navbar-right">
12.       (...)
13.     </ul>
14.   </div>
15. </nav>
```

Icons

We've included 360+ font icons.

```
1. <canvas width="150" height="150" class="chart" data-show-
2. stroke="true" data-percentage="50" data-middle-space="0" data-second-
3. color="#d30000" data-first-color="#aacd4a"></canvas>
```

Typography

We use google font service to include fonts or font face sets.

Serious Font used: Lato

Creative Font used: Gloria Hallelujah

JavaScript

All javascripts are placed in folder /js. In main.js file we implement all functions calls.

This theme uses following Javascript files:

1. Bootstrap v 3 framework (powerful front-end framework for faster and easier web development)
2. jQuery v1.10.2 (A fast javascript library that Foodtruckifies how to traverse HTML documents, handle events, perform animations.)
3. jQuery FlexSlider v2.2.0
4. GMAP3 Plugin for JQuery v. 5.1.1
5. jQuery.appear
6. FitVids 1.0.3
7. jQuery Stellar
8. jQuery tweet
9. Modernizr 2.0.6
10. PageScroller
11. final-countdown.js
12. snap.js
13. device.js 0.1.58
14. chart.js

Google Map (GMAP3 Plugin)

This is google map with custom marker, you can define couple parameters:

- **data-location** - address to show
- **data-height** - map height (if undefined height will be 220px)
- **data-offset** - by how many map should be repositioned from marker center point (default -30)

It uses gmap3 plugin.

Documentation: [here](#)

HTML Markup:

```
1. <div id="googleMap" class="googleMap" data-location="Seward Park" data-text=""></div>
```

All javascript code can be found in js/main.js file:

Flexslider

Content / Images Slider.

Each parameter can be defined as data attribute :

- data-namespace
- data-selector
- data-animation

- data-easing
- data-direction
- data-reverse
- data-animationloop
- data-smoothheight
- data-startat
- data-slideshow
- data-slideshowspeed
- data-animationspeed
- data-initdelay
- data-randomize
- data-thumbcaptions
- data-pauseonaction
- data-pauseonhover
- data-pauseinvisible
- data-usecss
- data-touch
- data-video
- data-controlnav
- data-directionnav
- data-prevtext
- data-nexttext
- data-keyboard
- data-multiplekeyboard
- data-mousewheel
- data-pauseplay
- data-pausetext
- data-playtext
- data-itemwidth
- data-itemmargin
- data-minitems
- data-maxitems
- data-move
- data-allowoneslide
- data-controlscontainer
- data-manualcontrols
- data-sync
- data-asnavfor

```
1.    /* ===== */
2.    /* ==== FLEXSLIDER ==== */
3.    if(jQuery().flexslider){
```



```
30.         var ctvideo = parseBoolean($this.attr("data-video"), false);
31.         var ctcontrolnav= parseBoolean($this.attr("data-controlnav"),
true);
32.         if(ctcontrolnav == false){
33.             $this.addClass("no-margin");
34.         }
35.         var ctdirectionnav= parseBoolean($this.attr("data-
directionnav"), false);
36.         var ctprevtext = validatedata($this.attr("data-
prevtext"),"Previous");
37.         var ctnexttext = validatedata($this.attr("data-
nexttext"),"Next");
38.         var ctkeyboard= validatedata(parseBoolean($this.attr("data-
keyboard")), true);
39.         var ctmultiplekeyboard= parseBoolean($this.attr("data-
multiplekeyboard"), false);
40.         var ctmousewheel= parseBoolean($this.attr("data-mousewheel"),
false);
41.         var ctpauseplay= parseBoolean($this.attr("data-pauseplay"),
false);
42.         var ctpausetext = validatedata($this.attr("data-
pausetext"),"Pause");
43.         var ctplaytext = validatedata($this.attr("data-
playtext"),"Play");
44.         var ctitemwidth= parseInt(validatedata($this.attr("data-
itemwidth"), 0));
45.         var ctitemmargin= parseInt(validatedata($this.attr("data-
itemmargin"), 0));
46.         var ctminitems= parseInt(validatedata($this.attr("data-
minitems"), 0));
47.         var ctmaxitems= parseInt(validatedata($this.attr("data-
maxitems"), 0));
48.         var ctmove= parseInt(validatedata($this.attr("data-move"),
0));
49.         var ctallowoneslide= parseBoolean($this.attr("data-
allowoneslide"), false);
50.         var ctcontrolscontainer= validatedata($this.attr("data-
controlscontainer"), "");
51.         var ctmanualcontrols= validatedata($this.attr("data-
manualcontrols"), "");
52.         var ctsync = validatedata($this.attr("data-sync"), "");
53.         var ctasnavfor = validatedata($this.attr("data-asnavfor"),
"");
54.         $this.flexslider({
```

```
55.         namespace: ctnamespace,           //{NEW} String: Prefix
string attached to the class of every element generated by the plugin
56.         selector: ctselector,           //{NEW} Selector: Must match a
simple pattern. "{container} > {slide}" -- Ignore pattern at your own peril
57.         animation: ctanimation,         //{String: Select your
animation type, "fade" or "slide"
58.         easing: cteasing ,               //{NEW} String:
Determines the easing method used in jQuery transitions. jQuery easing plugin
is supported!
59.         direction: ctdirection,         //{String: Select the
sliding direction, "horizontal" or "vertical"
60.         reverse: ctreverse,             //{NEW} Boolean:
Reverse the animation direction
61.         animationLoop: ctanimationloop, //Boolean:
Should the animation loop? If false, directionNav will received "disable"
classes at either end
62.         smoothHeight: ctsmoothheight,   //{NEW} Boolean:
Allow height of the slider to animate smoothly in horizontal mode
63.         startAt: ctstartat,             //{Integer: The
slide that the slider should start on. Array notation (0 = first slide)
64.         slideshow: ctslideshow,        //{Boolean: Animate
slider automatically
65.         slideshowSpeed: ctslideshowspeed, //Integer: Set
the speed of the slideshow cycling, in milliseconds
66.         animationSpeed: ctanimationspeed, //Integer:
Set the speed of animations, in milliseconds
67.         initDelay: ctinitdelay,         //{NEW} Integer:
Set an initialization delay, in milliseconds
68.         randomize: ctrandomize,        //{Boolean: Randomize
slide order
69.         thumbCaptions: ctthumbcaptions, //Boolean:
Whether or not to put captions on thumbnails when using the "thumbnails"
controlNav.
70.         // Usability features
71.         pauseOnAction: ctpauseonaction, //Boolean:
Pause the slideshow when interacting with control elements, highly
recommended.
72.         pauseOnHover: ctpauseonhover,   //{Boolean: Pause
the slideshow when hovering over slider, then resume when no longer hovering
73.         pauseInvisible: ctpauseinvisible, //{NEW}
Boolean: Pause the slideshow when tab is invisible, resume when visible.
Provides better UX, lower CPU usage.
74.         useCSS: ctusecss,               //{NEW} Boolean:
Slider will use CSS3 transitions if available
```

75. touch: cttouch, // {NEW} Boolean: Allow touch swipe navigation of the slider on touch-enabled devices

76. video: ctvideo, // {NEW} Boolean: If using video in the slider, will prevent CSS3 3D Transforms to avoid graphical glitches

77. // Primary Controls

78. controlNav: ctcontrolnav, // Boolean: Create navigation for paging control of each slide? Note: Leave true for manualControls usage

79. directionNav: ctdirectionnav, // Boolean: Create navigation for previous/next navigation? (true/false)

80. prevText: ctprevtext, // String: Set the text for the "previous" directionNav item

81. nextText: ctnexttext, // String: Set the text for the "next" directionNav item

82. // Secondary Navigation

83. keyboard: ctkeyboard, // Boolean: Allow slider navigating via keyboard left/right keys

84. multipleKeyboard: ctmultiplekeyboard, // {NEW} Boolean: Allow keyboard navigation to affect multiple sliders. Default behavior cuts out keyboard navigation with more than one slider present.

85. mousewheel: ctmousewheel, // {UPDATED} Boolean: Requires jquery.mousewheel.js (<https://github.com/brandonaaron/jquery-mousewheel>) - Allows slider navigating via mousewheel

86. pausePlay: ctpauseplay, // Boolean: Create pause/play dynamic element

87. pauseText: ctpausetext, // String: Set the text for the "pause" pausePlay item

88. playText: ctplaytext, // String: Set the text for the "play" pausePlay item

89. // Special properties

90. controlsContainer: ctcontrolscontainer, // {UPDATED} jQuery Object/Selector: Declare which container the navigation elements should be appended to. Default container is the FlexSlider element. Example use would be \$(".flexslider-container"). Property is ignored if given element is not found.

91. manualControls: ctmanualcontrols, // {UPDATED} jQuery Object/Selector: Declare custom control navigation. Examples would be \$(".flex-control-nav li") or "#tabs-nav li img", etc. The number of elements in your controlNav should match the number of slides/tabs.

92. sync: ctsync, // {NEW} Selector: Mirror the actions performed on this slider with another slider. Use with care.


```

93.         asNavFor: ctasnavfor,                               //{NEW} Selector:
Internal property exposed for turning the slider into a thumbnail navigation
for another slider
94.         // Carousel Options
95.         itemWidth: ctitemwidth,                             //{NEW} Integer:
Box-model width of individual carousel items, including horizontal borders and
padding.
96.         itemMargin: ctitemmargin,                           //{NEW}
Integer: Margin between carousel items.
97.         minItems: ctminitems,                               //{NEW} Integer:
Minimum number of carousel items that should be visible. Items will resize
fluidly when below this.
98.         maxItems: ctmaxitems,                               //{NEW} Integer:
Maximum number of carousel items that should be visible. Items will resize
fluidly when above this limit.
99.         move: ctmove,                                       //{NEW} Integer:
Number of carousel items that should move on animation. If 0, slider will move
all visible items.
100.        allowOneSlide: ctallowoneslide                       //{NEW} Boolean:
Whether or not to allow a slider comprised of a single slide
101.        })
102.    })
103.    }

```

jquery.appear

This plugin call function when element appear in viewport. This is used in custom animation, progress bars, counter timer.

Documentation: <https://github.com/bas2k/jquery.appear/>

```

1.    if (jQuery().appear) {
2.        if (device.mobile()) {
3.            // disable animation on mobile
4.            jQuery("body").removeClass("withAnimation");
5.        } else {
6.            jQuery('.withAnimation .animated').appear(function () {
7.                jQuery(this).each(function () {
8.                    jQuery(this).addClass('activate');
9.                    jQuery(this).addClass($(this).data('fx'));
10.                });
11.            }, {accY: -150});
12.        }
13.    }

```

fitVids

Just make video responsive.

Documentation: <http://fitvidsjs.com/>

HTML Markup:

```
1. <div class="fit-video">
2.   <iframe width="853" height="480" src="//www.youtube.com/embed/HJ2F7eptn_A?
   rel=0" frameborder="0" allowfullscreen></iframe>
3. </div>
```

and js:

Charts.js

This plugin is used to make graphics.

Documentation: <http://www.chartjs.org/>

Contact Form

Template comes with ready to use:

- Contact Form
- Newsletter Form

All needed files are in /form directory.

Remember to include `<script src="form/js/form.js"></script>` on page.

To **configure** form just edit form/form.php and change e-mails to your own :

```
1. $to = "mail@example.com";
2. $from = "mail@example.com";
```

That's it!

Contact Form

HTML Markup:

```
1. <h5 class="light pull-right">Send a Message</h5>
2. <div class="row padding-xs-top">
3.   <div class="successMessage alert alert-success" style="display: none">
4.     <button type="button" class="close" data-dismiss="alert" aria-
```

```
hidden="true"></button>
5.         <p>Thank You! We will contact you shortly.</p>
6.         </div>
7.         <div class="errorMessage alert alert-error" style="display: none">
8.             <button type="button" class="close" data-dismiss="alert" aria-
hidden="true"></button>
9.             <p>Ups! An error occured. Please try again later.</p>
10.        </div>
11.    </div>
12.    <form role="form" action="form/send.php" method="post" class="contactForm
validateIt" data-email-subject="Contact Form" data-show-errors="true">
13.        <div class="row padding-xs-top">
14.            <div class="col-md-6 col-sm-6">
15.                <div class="form-group">
16.                    <input required type="text" name="field[]" class="form-
control input-lg" placeholder="Name *">
17.                </div>
18.                <div class="form-group">
19.                    <input required type="tel" name="field[]" class="form-
control input-lg" placeholder="Phone">
20.                </div>
21.            </div>
22.            <div class="col-md-6 col-sm-6">
23.                <div class="form-group">
24.                    <input required type="email" name="field[]" class="form-
control input-lg" placeholder="Email *">
25.                </div>
26.                <div class="form-group">
27.                    <input required type="tel" name="field[]" class="form-
control input-lg" placeholder="Fax">
28.                </div>
29.            </div>
30.        </div>
31.        <div class="row">
32.            <div class="col-xs-12 col-sm-12 col-md-12">
33.                <div class="form-group">
34.                    <textarea class="form-control input-lg" rows="4"
name="field[]" required
35.                        placeholder="Your message"></textarea>
36.                </div>
37.            </div>
38.        </div>
39.        <div class="row">
40.            <div class="col-xs-12">
```

```

41.         <button type="submit" class="btn btn-primary btn-lg pull-
right">Send Message
42.         </div>
43.     </div>
44. </form>

```

How to add more fields to form ?

Just add another form-group section,
for example additional phone and fax :

```

1.     <div class="form-group">
2.         <label>Your Phone</label>
3.         <input type="text" class="form-control" name="field_[]"
placeholder="enter your phone">
4.     </div>
5.     <div class="form-group">
6.         <label>Your Fax</label>
7.         <input type="text" class="form-control" name="field_[]"
placeholder="enter your fax">
8.     </div>

```

Newsletter Form

HTML Markup:

```

1.     <h4 class="uppercase motive section-top">Subscribe to Newsletter</h4>
2.     <br>
3.     <div class="successMessage alert alert-success" style="display: none">
4.         <button type="button" class="close" data-dismiss="alert" aria-
hidden="true">x</button>
5.         Thank You!
6.     </div>
7.     <div class="errorMessage alert alert-danger" style="display: none">
8.         <button type="button" class="close" data-dismiss="alert" aria-
hidden="true">x</button>
9.         Ups! An error occured. Please try again later.
10.    </div>
11.    <form class="form-inline validateIt" role="form"
action="../../assets/form/send.php"
12.        method="post" data-email-subject="Newsletter Form" data-show-
errors="true"
13.        data-hide-form="true">
14.        <div class="input-group">

```

```

15.         <input type="email" name="field[]" class="form-control"
16.             placeholder="Enter your email address">
17.         <input type="hidden" name="msg_subject" value="New Newsletter
subscription">
18.             <span class="input-group-btn">
19.                 <button class="btn btn-brick" type="submit">GO</button>
20.             </span>
21.         </div>
22.     </form>

```

- **Mail Subject** - just change value in : <input type="hidden" name="msg_subject" value="Contact Form">
- **Success Message** - change successMessage div content
- **Error Message** - change errorMessage div content

Twitter Feed

This is live twitter feed. This plugin will grab your latest tweets and display them on page.

Configuration: - just edit twitter/config.php file :

```

1.  $config = array(
2.      'username' => "XXX", //username
3.      'limit' => 3, //how many tweets we would like to display?
4.      'with_replies' => true, //show replies? true or false
5.      'oauth_access_token' => 'TOKEN',
6.      'oauth_access_token_secret' => 'TOKEN_SECRET',
7.      'consumer_key' => 'KEY',
8.      'consumer_secret' => 'KEY_SECRET',
9.  );

```

Where can I get these keys, tokens, etc ?

Twitter API requires authentication to display tweets from your account.

To twitter feed work you need to create the 'application'

and generate 4 special keys for your twitter account via <https://dev.twitter.com/apps> :

- Consumer key (key)
- Consumer secret (secret)
- Access token (token)
- Access token secret (token_secret)

Video tutorial : "How to create Twitter App"

<http://vimeo.com/60891535>

Sources & Credits

The following sources and files have been used to build up this theme

Once again, thank you so much for purchasing this theme. If you have any questions, feel free to email me via my user page [contact form](#).

Copyright YourName 2014 made with the [Documenter v2.0](#)