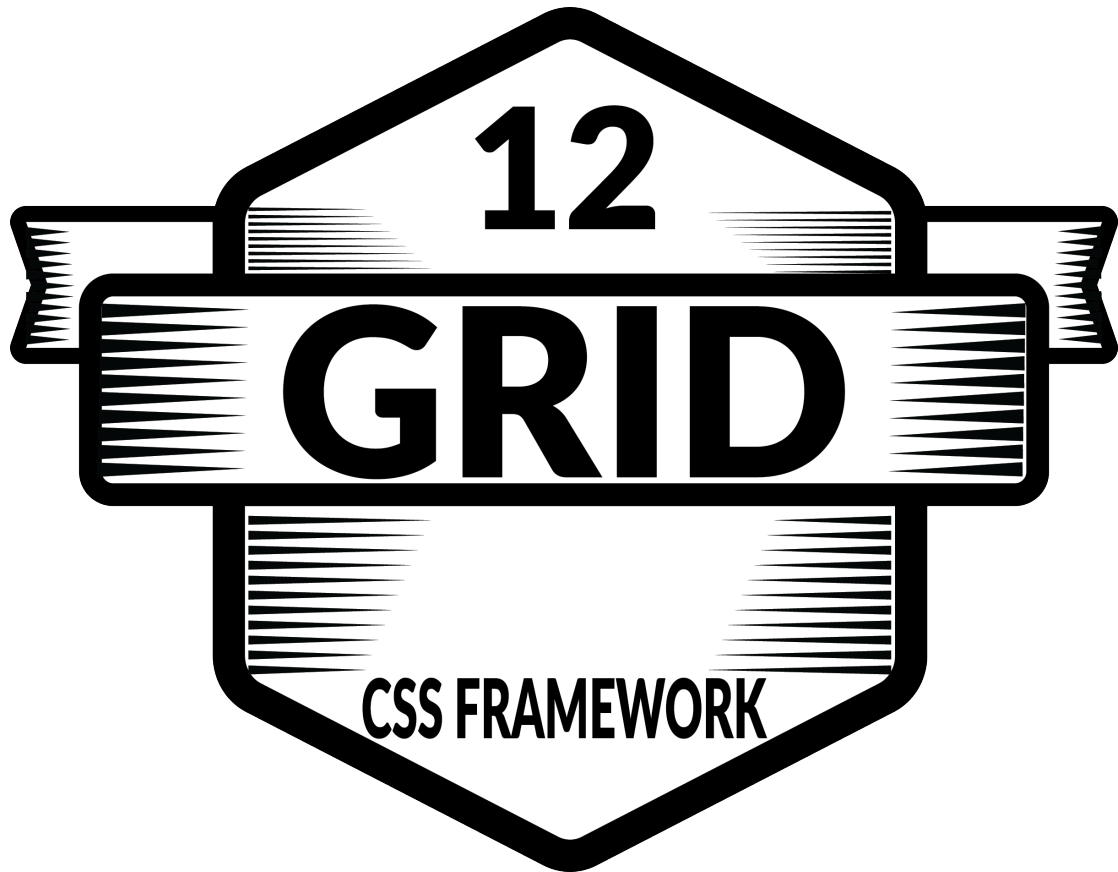# Grid 12 CSS Framework.
User Manual/Documentation.

Introduction:

Grid 12

This Framework was created by James Fletcher for a University of Salford Dissertation Piece.
More information at: www.grid12.com
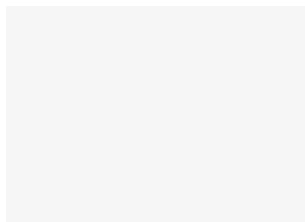Please use the example site found at www.grid12.com as a reference.
Thank you.
James Fletcher.

Glossary

Table of Contents

- Base Styles

Background colour defaulted at f6f6f6 (off white-grey).

In addition the framework also features.

- Grid

Grid 12 uses a 12-column system designed to give the user flexibility and accurate positioning classes. Each column is a percentage of 100% with a maximum width of 970px with equates to a typical desktop viewing width. The minimum size the framework will function at is larger than 550px smaller then

that the framework will condense to a vertically stacked system this avoids collision errors and maximizes reading size in a mobile resolution.

ONE	ELEVEN
TWO	TEN
THREE	NINE
FOUR	EIGHT
FIVE	SEVEN
SIX	SIX
SEVEN	FIVE
EIGHT	FOUR
NINE	THREE
TEN	TWO
ELEVEN	ONE
TWELVE	

Classes/

```
.col-1{ width: 4.66666666667%; }
  .col-2{ width: 13.3333333333%; }
  .col-3{ width: 22%;  }
  .col-4{ width: 30.6666666667%;     }
  .col-5{ width: 39.3333333333%; }
  .col-6{ width: 48%;   }
  .col-7{ width: 56.6666666667%;  }
  .col-8{ width: 65.3333333333%; }
  .col-9{ width: 74.0%;  }
  .col-10{ width: 82.6666666667%;   }
  .col-11{ width: 91.3333333333%; }
  .col-12{ width: 96%; margin-left: 0; }
```

100\12=4.66666 and so on. Give or take some guttering for collision errors and aesthetic.
The framework also includes a "full-col" class, which function as a 100%. In addition to a half and 1/3 classes for easy alignment.

```
  .half-col{ width: 50%;   }
   .full-col{ width: 100%;   }
    .col-1-3          { width: 33.33333333333333%; }
```

In addition to the standard grid grid12 also includes a margin-left functioning "push-col" class as an alternative to help supply the user with more options.

```
 .push-col-1{ margin-left: 8.66666666667%; }
 .push-col-2{ margin-left: 17.3333333333%; }
 .push-col-3{ margin-left: 26%;            }
 .push-col-4{ margin-left: 34.6666666667%; }
 .push-col-5{ margin-left: 43.3333333333%; }
 .push-col-6{ margin-left: 52%;            }
 .push-col-7{ margin-left: 60.6666666667%; }
```

```
    .push-col-8{ margin-left: 69.3333333333%; }
    .push-col-9{ margin-left: 78.0%;          }
    .push-col-10{ margin-left: 86.6666666667%; }
    .push-col-11{ margin-left: 95.3333333333%; }
push
```

container = Basic container styling for an open page website. Sets width to a comfortable 960px (standard desktop size). While also giving you a padding of 20px.

row = Can be used as an alternative to the container class.

Guttering
This can be altered for the users need just be careful, as grid will have to be recalculated.

```
    [class*='col-'] {
        float: left;
        min-height: 1px;
padding: 1px;
    }
```

- Typography

Type is an important factor in any framework. We felt that Lato was a good option for the framework. All headers condense to phablet and below as well as cascade well in scale.

http://fonts.googleapis.com/css?family=Lato&subset=latin,latin-ext

Basic heading styling can be altered to suit user. Fully collapsible and works excellently in mobile view.

```
119  /* Typography
120  ---------------------------------------------------- */
121  h1, h2, h3, h4, h5, h6 {
122    margin-top: 0;
123    margin-bottom: 2rem;
124    font-weight: 300; }
125  h1 { font-size: 4.0rem; line-height: 1.2;  letter-spacing: -.1rem;}
126  h2 { font-size: 3.6rem; line-height: 1.25; letter-spacing: -.1rem; }
127  h3 { font-size: 3.0rem; line-height: 1.3;  letter-spacing: -.1rem; }
128  h4 { font-size: 2.4rem; line-height: 1.35; letter-spacing: -.08rem; }
129  h5 { font-size: 1.5rem; line-height: 1.6;  letter-spacing: 0; }
130
131  /* Larger than phablet */
132  @media (min-width: 550px) {
133    h1 { font-size: 5.0rem; }
134    h2 { font-size: 4.2rem; }
135    h3 { font-size: 3.6rem; }
136    h4 { font-size: 2.4rem; }
137    h5 { font-size: 1.5rem; }
138  }
139  .p {
140    margin-top: 0; }
```

- Buttons

The button classes include a number of different colours. Also it includes two sizes small and larger for different user purposes.

A default button tag will look like this. No hover effects.
<button></button>

BUTTON ELEMENT

Can add class definitions to call on more size control and colour.

Example: class="l-button button-default"

Result:

BUTTON ELEMENT

(small colour hover effect)

Button – Big button class
L-button - Meaning Little button default colour.
Button-default – Color class can be changed to a number of different options colour definitions below.

Colour summary
Default - Grey
Primary – Blue
Success - Green
Danger - Red
Stylish - Pink

Classes examples
  .button (large/default button)
.l-button (small button)
.button button-default (large grey colour button)
.l-button button-default (Small grey colour button)
button-default
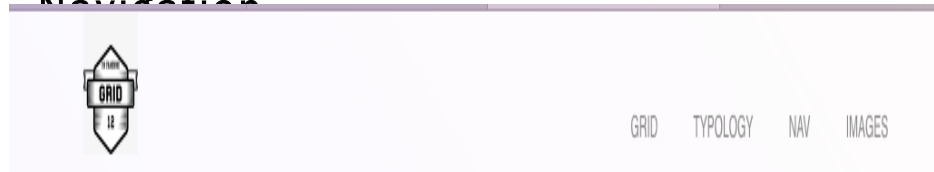

(Primary class buttons.)

- Featurette

The hero unit is used to give the opening of your website a simple and elegant introduction to your site. Its designed to work well with the Navbar and text and layout class can be used to display text and items within the unit.
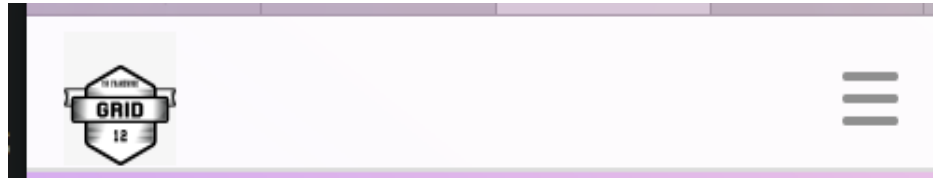

We used a simple gradient effect as an example but you could also use an image.
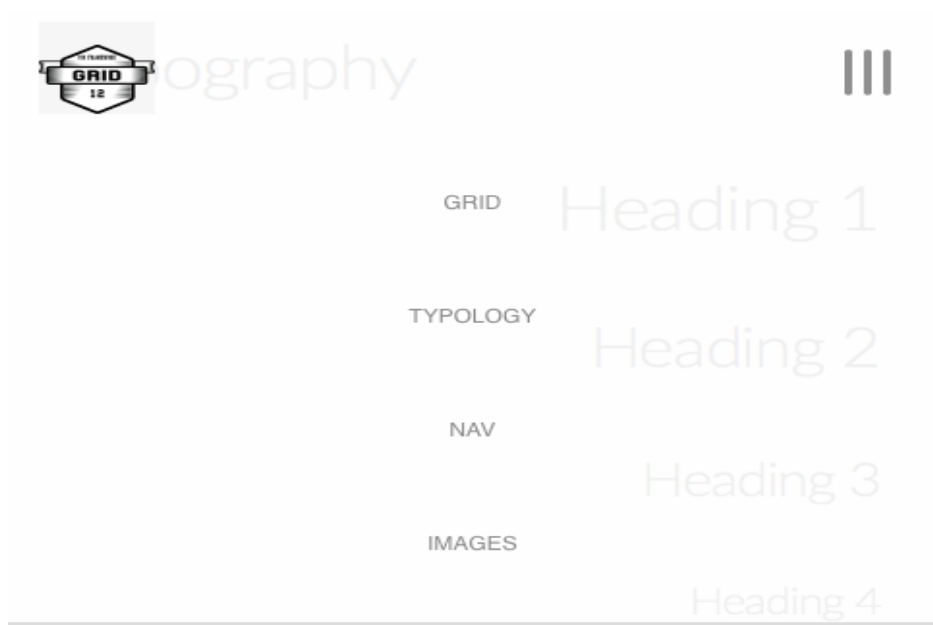
Navigation

Standard resolution.

Mobile View

Cascading mobile view.

Simple "sticky" Navbar appears after scroll with in page link scrolling effect and mobile view options. Customize the colours using the grid12.js file under navigation header. You can also change the scroll appearance height under by changing 100 to any number you want by default it is set to 100px. Change to 0 for permanent viewing without scrolling.

if ($(window).scrollTop() > 100) {

```
$(window).scroll(function() {

    if ($(window).scrollTop() > 100) {
        $('.main_h').addClass('sticky');
    } else {
        $('.main_h').removeClass('sticky');
    }
});
```

Colour class names summary:
.sticky - grey
.sticky-dark
.sticky-pink
.sticky-green
.sticky-blue

Logo or Text:
You can also decide weather to use your logo or just a text header but changing between class="logo" and class="logo-img". The logo is sized for the Navbar and should fit snuggly in the middle.

```
    <header class="main_h">
  <div class="row">
        <a href="#"> <img class="logo-img" src="css/img/grid12.png"></a>
        <div class="mobile-toggle">
            <span></span>
            <span></span>
            <span></span>
        </div>
        <nav>
            <ul>
                <li><a href="#section-one">Grid</a></li>
                <li><a href="#section-two">Typology</a></li>
                <li><a href="#section-three">Nav</a></li>
                <li><a href="#section-four">Images</a></li>
            </ul>
        </nav>
  </div> <!-- / row -->
```

Scrolling links:
Change using the in page JavaScript.

```
$.scrollTo = $.fn.scrollTo = function(x, y, options){
    if (!(this instanceof $)) return $.fn.scrollTo.apply($('html, body'), arguments);

    options = $.extend({}, {
        gap: {
            x: 0,
```

```
        y: -80
      },
      animation: {
        easing: 'swing',
        duration: 600,
        complete: $.noop,
        step: $.noop
      }
  }, options);
  return this.each(function(){
    var elem = $(this);
    elem.stop().animate({
      scrollLeft: !isNaN(Number(x)) ? x : $(y).offset().left + options.gap.x,
      scrollTop: !isNaN(Number(y)) ? y : $(y).offset().top + options.gap.y
    }, options.animation);
  });
};
```

Can be replaced with just standard page links.

- Lists

Line and unordered list styles.

Unordered lists:

<ul>
*   Lightly styled unordered lists
*
*   *
*   They use the square bulletpoint style*
*   *
*   Additional nested lists
*
*   Can nest either type of list into the other
*
*   *
*   *

```
        <li>Just another list example</li>
        </ul>
```

Ordered lists:

1. Lightly styled ordered lists

2. Ordered lists given decimal characters.

    ■ Ordered and unordered are both nestable

    ■ Can nest either type interchangably.

3. Yet another list example

```
<ol>
        <li>Lightly styled ordered lists</li>
        <li>
          Ordered lists given decimal characters.
          <ul>
            <li>Ordered and unordered are both nestable</li>
            <li>Can nest either type interchangably.</li>
          </ul>
        </li>
        <li>Yet another list example</li>
        </ol>
```

Enables the user to create indented paragraphs and headings as well as numbered lists and sub-lists.

Nothing exciting but useful nonetheless.

- Tables

Basic table Structure 4 columns extendable to any need.
Headers in bold and dotted lined <hr> tags.

Brand	Colletion	Size	Price
John Smith	Summer	Medium	¬£10.00
Steve Buscemi	Fall	Small	¬£15.00

```html
<table class="col-12">
  <thead>
   <tr>
     <th>Brand</th>
     <th>Colletion</th>
     <th>Size</th>
     <th>Price</th>
   </tr>
  </thead>
  <tbody>
   <tr>
     <td>Dave Gamache</td>
     <td>Summer</td>
     <td>Medium</td>
     <td>£10.00</td>
   </tr>
   <tr>
     <td>Dwayne Johnson</td>
     <td>Fall</td>
     <td>Small</td>
     <td>£15.00</td>
   </tr>
  </tbody>
</table>
```

- ## Image

Hover effect for button can overwritten based off user needs.
Minimum styling.

```css
img {
width: 100%;
height: auto;
margin: 0 auto;
}
img:hover {
  opacity: 0.9;
}
```

The class "max" just centers and maximizes imgs within a wrapper or container.

```
img .max{
    max-width: 100%;
    max-height: 100%;
    vertical-align: middle;
}
```

- Layout

Basic layout styling "full-width" used to absolute center a section or piece of html for mainly used in conduction with the hero unit for centralised text and buttons.

```
.full-width {
  width: 100%;
  box-sizing: border-box; }
.text-center-screen{
    position: absolute;
    width: 100px;
    height: 50px;
    top: 50%;
    left: 50%;
    margin-left: -50px;
    margin-top: -25px;
}
```

Vertical centered text used in hero unit and can be used in a number of different applications.

```
.text-vertical-center {
    text-align: center;
    vertical-align: text-middle;
}
```

- Modal forms

Basic email/message form doesn't include PHP although can be written in for the user. Follows the style of the framework. Includes a JavaScript anti spam question.

Feature explanation:

Modal Header ✕

👤 Name

✉ Email

✏ Comment

💬 2+2=

SEND

Can be used in conjunction with a modal function to save on space.
Simply add within the modal html structure this is shown on page #.