www.cocobasic.com

https://cocobasic.ticksy.com/

cocobasicthemes@gmail.com

Template Manual

Caliris HTML

Thank You :)

Thanks for purchasing template and for supporting our work. Caliris is our new fancy template. Responsive, clean and professional look with sleek design will satisfy all your needs.

You will need basic HTML, CSS, jQuery skills to edit Caliris HTML.
We will try to help you out with this documentation. If we fail here, you can always contact us on https://cocobasic.ticksy.com/ and our developers will try to solve your issue.

HTML Structure

Caliris consist several HTML files

- index.html - Front (Big) page
- single.html
- page.html
- blog.html
- portfolio-1.html
- portfolio-2.html
- portfolio-3.html
- team-1.html
- team-2.html
- team-3.html
- team-4.html
- team-5.html
- team-6.html

Front Page index.html

On Home page are placed 9 sections:

Home, Services, Team, Skills, Video, News, Portfolio, About and **Clients.** Each section is separated and start with code like this:

```
<!-- Home Section -->
        <div id="home" class="section full-width full-screen">

                ...

        </div>


<!-- Services Section -->
        <div id="services" class="section page-split">

                ...

        </div>



    <!-- Team Section -->
        <div id="team" class="section page-split">

                …

        </div>
```

Home Section

This section start with code like this:

```
<div id="home" class="section full-width full-screen">
```

Here is important class "**full-screen**" - this class will set the height of section (it this case Home section) to be minimal height as device/browser height. So, it will "stretch" to fit the full screen.

This class we are using only on Home section.

Also, on Home section is placed some text and button. Code for button looks like this:

```
<h1 class="big-text ">
        We Craft Awesome <br>
        Web And Graphic Design <br>
        Solutions For You
</h1>
<div class="text-left">
        <a href="#footer" class="button scroll top-30">Get in Touch</a>
</div>
```

This button is linking to **Footer** section (**href="#footer"**) and there is a class "**scroll**" which is important - if you don't have this class, there will not be smooth scroll animation. So, if you want to link to any section inside page and to have smooth scroll animation, you need to include "**scroll**" class in link (<a> tag).

Service Section

This section start with code like this:

```
<div id="services" class="section page-split">
```

Important class here is "**page-split**" - this is the mean the section will have a layout where title is placed in one part of screen (left/right) and the content will be placed in another part of screen.

The next important part on this section is this code:

```
<div class="bg-holder float-left">

    <img src="images/left_obj_01.png" alt="">

</div>

<div class="section-title-holder float-left" data-background="#a13dd7">
```

Here is important class "**float-left**" with this class you can set the title to be on left part of screen or with class "**float-right**" to be on right part of screen.

Another important part is this code **data-background="#a13dd7"** - with this code you will set the background color of sections on small resolution when the image placed in "**bg-holder**" (which is a background image for section title on big resolution) is hidden.

Also, on Service section you will find code like this for service item:

```
<div class="one_third animate wait-07s">

            <div class="service-item">

                <img class="service-icon" src="images/icon_01.png" alt="">

                <p class="service-title">Premium</p>

                <div class="service-content">

                    Cosmic ocean another world the only home we've ever known

                </div>

            </div>

        </div>
```

On Caliris HTML are several elements which have fadeIn animation when the element comes in viewport (one example is above for service item). To add element animation, all you need to do is to add class "**animate**". For example:

```
<div class="one_third animate">
```

Also, there are classes which allows you to set 0.1s - 1.5s animation delay ("**wait-01s**", "**wait-02s**" … "**wait-14s**", "**wait-15s**"). Use it like this:

```
<div class="one_third animate wait-07s">
```

Team Section

Here you will find code for team member:

```
<li class="member-holder one_third">

        <a class="img-link ajax-member-content" href="team-1.html" data-id="1">

                <img src="images/team_01.jpg" alt="">

                <div class="member-mask">

                        <img src="images/icon_plus.png" alt="">

                </div>

        </a>

        <p class="member-name">Mark Stone</p>

        <p class="member-position">CEO</p>

</li>
```

Team member content is loaded in page using Ajax. The important part here is this line of code:

```
<a class="img-link ajax-member-content" href="team-1.html" data-id="1">
```

Class "**ajax-member-content**" must be included if you want to have in page/ajax load. If you remove this class, the link will work on "regular way" and will open the page which you have set in "**href**". Another important part, is to set unique **data-id** number of each member like: **data-id="1" , data-id="2" , data-id="3" ….**

So, when you click on member image, the ajax will load the content from html file which is set in "**href**" attribute (in example above it is "**team-1.html**" file).

The ajax will load a whole file, but we are displaying only content which is placed inside this div:

```
<div class="member-item-wrapper center-relative">

…

</div>
```

All content which is placed inside this div will be loaded and displayed on front/home page (index.html) using Ajax.

Skills Section

Here you will find code for skills:

```html
<div class="v-skills-holder">

    <div class="v-skill">

        <div class="v-skill-fill"></div>

        <div class="v-skill-text">

            <span data-color="#27E0C0">75%</span>

            Creativity

        </div>

    </div>

    <div class="v-skill">

        <div class="v-skill-fill"></div>

        <div class="v-skill-text">

            <span data-color="#27E0C0">20%</span>

            Cooking

        </div>

    </div>

    <div class="v-skill">

        <div class="v-skill-fill"></div>

        <div class="v-skill-text">

            <span data-color="#27E0C0">90%</span>

            Development

        </div>

    </div>

</div>
```

Video Section

Here you will find code for video popup:

```
<a class="video-popup-holder" href="https://vimeo.com/150685211" data-rel="prettyPhoto[gallery-video1]">

<p class="popup-play"></p>

</a>
```

You can set a Vimeo or YouTube video in popup, but you must use this format:

YouTube video in format:

http://www.youtube.com/watch?v=XXXXXX

or Vimeo video in format:

http://vimeo.com/XXXXXX

Portfolio Section

Here we have 2 different ways how we can link portfolio items. On our demo, the first 3 items are linked via ajax and load content inside the home/front page.

The 4rth and 5th item is linking to popup image and video.

Here is a example for portfolio item which is loading content via ajax:

```html
<div id="p-item-1" class="grid-item element-item p_one">

        <a class="item-link ajax-portfolio" href="portfolio-1.html" data-id="1">

                <img src="images/item_01.jpg" alt="">

                <div class="portfolio-text-holder">

                        <p class="portfolio-title">Plate</p>

                        <p class="portfolio-desc">REAL MOCKUP</p>

                </div>

                <p class="portfolio-plus">

                        <img src="images/icon_plus.png" alt="">

                </p>

        </a>

</div>
```

Portfolio item content is loaded in page using Ajax. The important part here is this 2 lines of code:

```
<div id="p-item-1" class="grid-item element-item p_one">

        <a class="item-link ajax-portfolio" href="portfolio-1.html" data-id="1">
```

Class "**ajax-portfolio**" must be included if you want to have in page/ajax load. If you remove this class, the link will work on "regular way" and will open the page which you have set in "**href**". Another important part, is to set unique **data-id** number of each member like: **data-id="1" , data-id="2" , data-id="3" ….**

Also, the **id="p-item-1"** must be unique for each item.

So, when you click on portfolio item image, the ajax will load the content from html file which is set in "**href**" attribute (in example above it is "**portfolio-1.html**" file).

The ajax will load a whole file, but we are displaying only content which is placed inside this div:

```
 <div class="portfolio-item-wrapper center-relative">

…

</div>
```

All content which is placed inside this div will be loaded and displayed on front/home page (index.html) using Ajax.

The another way of linking portfolio items is to link to image/video popup video (4rth and 5th items in portfolio on our demo). Here is example:

```
<div class="grid-item element-item p_one">

        <a class="item-link" href="images/item_04.jpg" data-rel="prettyPhoto[portfolio]">

                <img src="images/item_04.jpg" alt="">

                <div class="portfolio-text-holder">

                        <p class="portfolio-title">Tower</p>

                        <p class="portfolio-desc">SKYSCRAPER</p>

                </div>

                <p class="portfolio-plus">

                <img src="images/icon_plus.png" alt=""></p>

        </a>

</div>


<div class="grid-item element-item p_one">

<a class="item-link" href="https://www.youtube.com/watch?v=15cpIHjEsWI" data-rel="prettyPhoto[portfolio]">

        <img src="images/item_05.jpg" alt="">

        <div class="portfolio-text-holder">

                <p class="portfolio-title">A4 Paper</p>

                <p class="portfolio-desc">PSD MOCKUP</p>

        </div>

        <p class="portfolio-plus">

                <img src="images/icon_plus.png" alt="">

        </p>

</a>

</div>
```

Single post - single.html

On Single post you will find code like this (image slider code):

```
<script>

    var slider1_speed = "2000";

    var slider1_auto = "true";

    var slider1_hover = "true";

    var slider1_items = "1";

</script>

<div class="image-slider-wrapper relative">

    <div id="slider1" class="owl-carousel owl-theme image-slider slider">

        <div class="owl-item">

            <img src="images/blog_img_large1.jpg" alt="">

        </div>

        <div class="owl-item">

            <img src="images/blog_img_large2.jpg" alt="">

        </div>

        <div class="owl-item">

            <img src="images/blog_img_large3.jpg" alt="">

        </div>

    </div>

</div>
```

If you need one more slider, duplicate that code and change "**slider1**" string on several places with new string, for example "**slider2**". So you will have "**slider2_speed**", "**slider2_auto**", id="**slider2**"...

Source & Credits

Special thanks to creators and contributors of these awesome libraries, we couldn't done this without them.

Images used in theme

https://unsplash.com

VIdeo used in template

https://www.youtube.com/watch?v=15cpIHjEsWI

https://vimeo.com/199192931

jQuery Isotope Plugin

http://isotope.metafizzy.co

OwlCarousel2

https://owlcarousel2.github.io/OwlCarousel2/

HTML5 Fallback Support

https://code.google.com/p/html5shiv

Respond JS

https://github.com/scottjehl/Respond

jQuery

http://jquery.com

Google Web Fonts

http://www.google.com/webfonts

Easing jQuery Plugin

http://gsgd.co.uk/sandbox/jquery/easing

Smart Menus

http://www.smartmenus.org/

Sticky

http://stickyjs.com/

Font Awesome

http://www.fontawesome.io/

FitVideo.js

http://fitvidsjs.com/

PrettyPhoto

http://www.no-margin-for-errors.com/

We are sorry if we forgot to mention someone.

Questions?

We tried to help you out with this documentation. So please read carefully. If we fail here, you can always go to our [support forum](#) and post a new discussion in appropriate category. Our developers will try to solve your issue.

Thanks for understanding.

www.cocobasic.com

https://cocobasic.ticksy.com/

cocobasicthemes@gmail.com