

Your assignment this week is to create a web-based self portrait. Try to think of things about or ways of expressing yourself that aren't already printed + categorized online.

The website must be at least **3 HTML** pages with **2 CSS** styling. At least 1 .css file that controls multiple pages (and inline CSS if you prefer for the 2nd design). The finished project should be sftp'd to your storm server for Monday's class.

For Thursday: create a sitemap and wireframe for your site. Post pictures of your process.



Parmigianino

Self-Portrait in a Convex Mirror c. 1524



Ana Mendieta
Imagen de Yagul, 1973



Pipiloti Rist, born 1962
Ever is Overall, 1997



Login fmfaculty.hunter.cuny.edu

fmundergrad@hunter.cuny.edu – SFTP with username
and password.

Username: leopoldr

Password: [REDACTED]

Anonymous Login

SSH Private Key: None

Add to Keychain



Cancel

Login

leopoldr@fmfaculty.hunter.cuny.edu – SFTP

Unregistered

FERED

Open Connection Quick Connect Action Refresh Edit

/Volumes/faculty/Faculty_accts3/leopoldr

Filename	Size	Modified
Sites		-- 9/12/18, 5:51 PM
week2		-- 9/12/18, 9:41 PM
week1		-- 9/12/18, 8:10 PM
Streaming		-- 8/27/18, 9:46 AM
index.html	2.6 KB	9/12/18, 8:05 PM
images		-- 8/27/18, 9:46 AM
em2		-- Today, 11:59 AM
Public		-- 8/27/18, 9:46 AM
Pictures		-- 8/27/18, 9:46 AM
Music		-- 8/27/18, 9:46 AM
Movies		-- 8/27/18, 9:46 AM
Library		-- 9/12/18, 5:38 PM
Downloads		-- 1/28/19, 12:01 PM
Documents		-- 1/28/19, 11:15 AM
Desktop		-- 1/28/19, 12:03 PM
Applications		-- 8/29/18, 5:42 PM

rebecca (marks) leopold

- [Week 01](#)
- [Week 02](#)

Elements Console

```
<!doctype html>
<html style="visibility: visible;">
  <head>...</head>
  ...<body> == $0
    <h4>rebecca (marks) leopold</h4>
      <ul>...</ul>
    </body>
</html>
```

<a> relative urls

Linking to pages on the same site

Parent Directory Home Page

Same Directory Week 1 Page

Child Directory Week 1 Page

id attribute Click here for this week in potatoes!

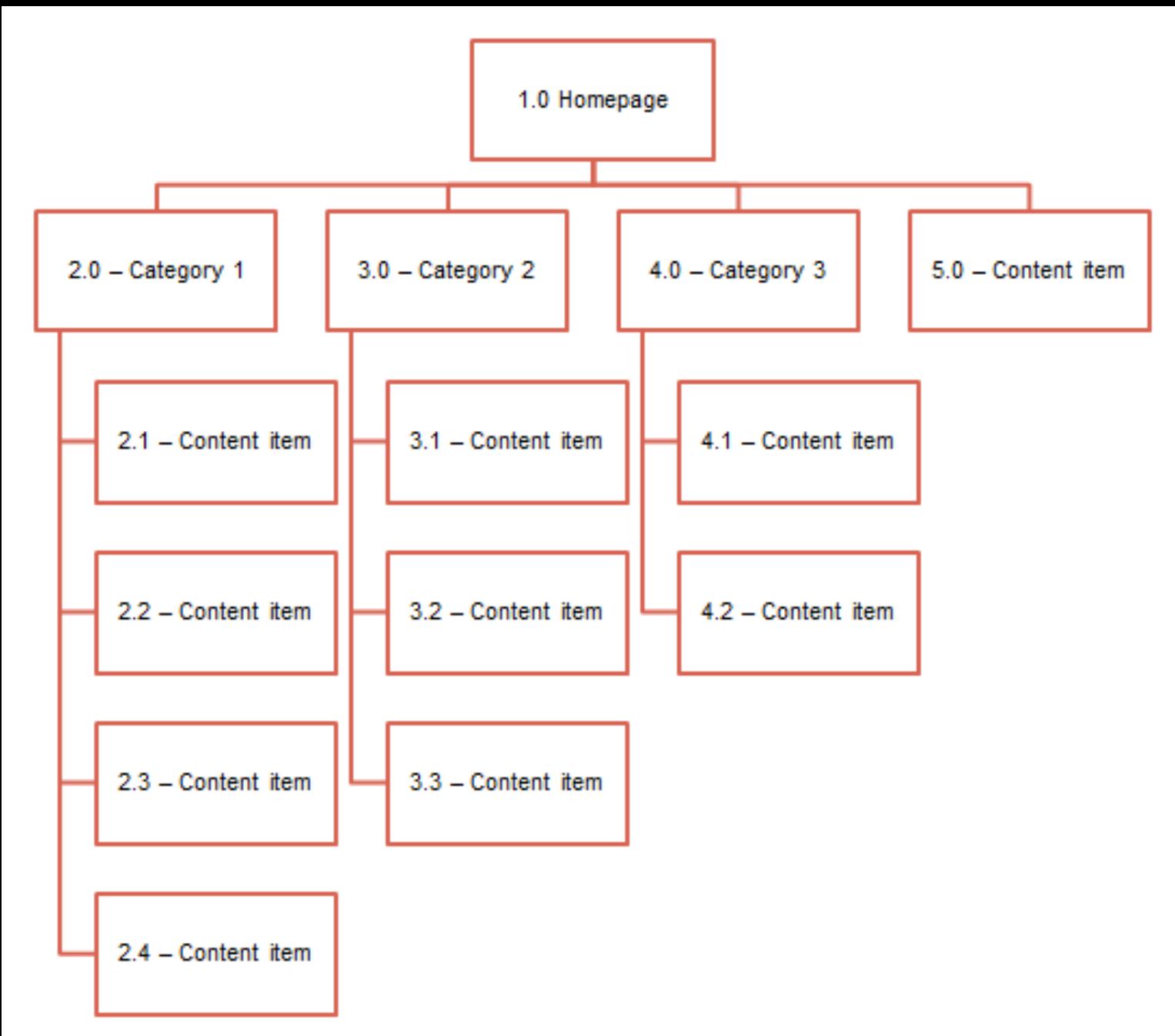
< a href — stands for *hyperlink reference*

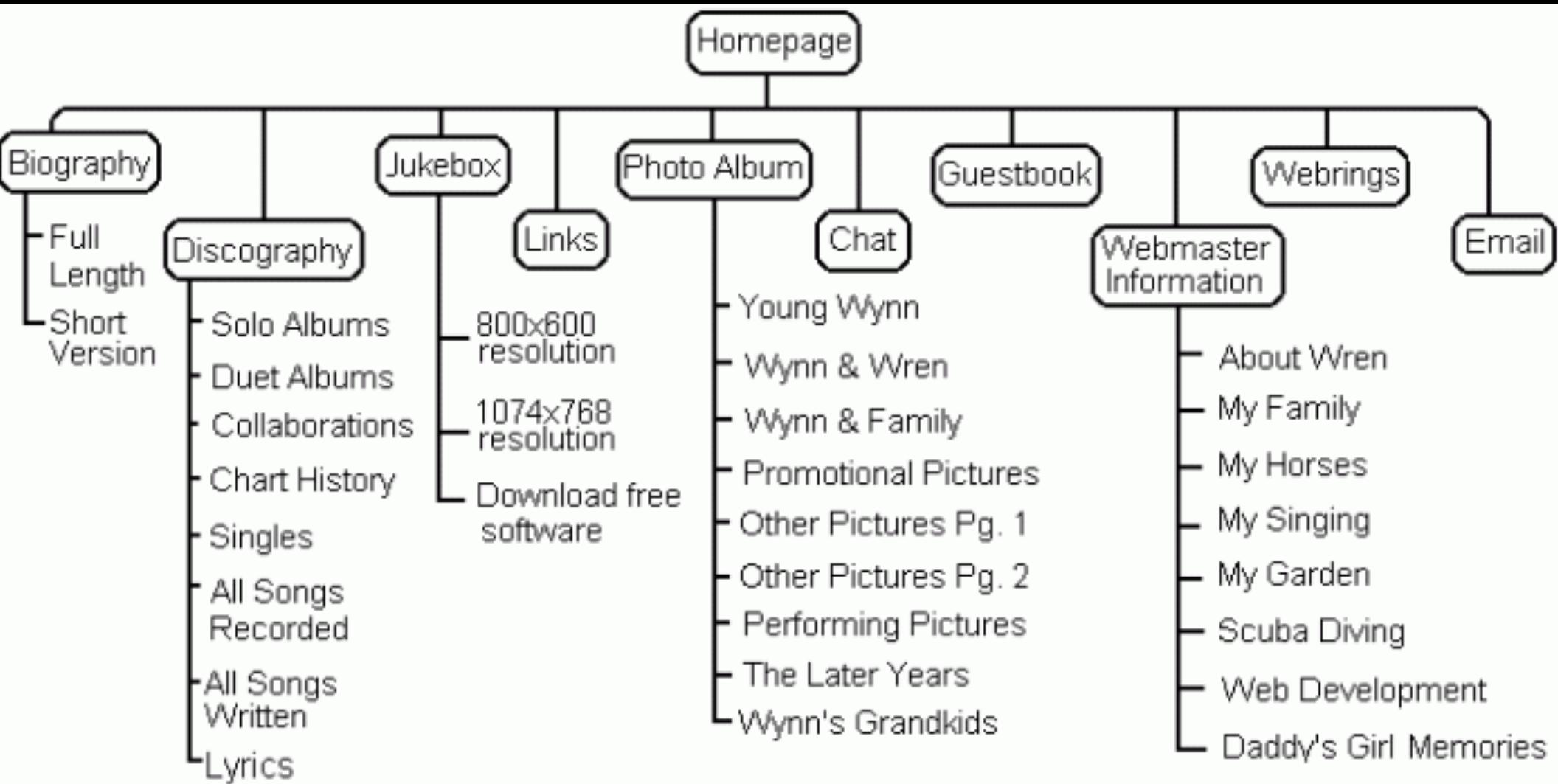
Why index.html?

The main homepage of a site written in HTML (and the homepage of each section in a child folder) is called index.html.

Web servers are usually set up to return the index.html file if no file name is specified. Therefore, it's always a good idea to name your directories' root webpages index.html

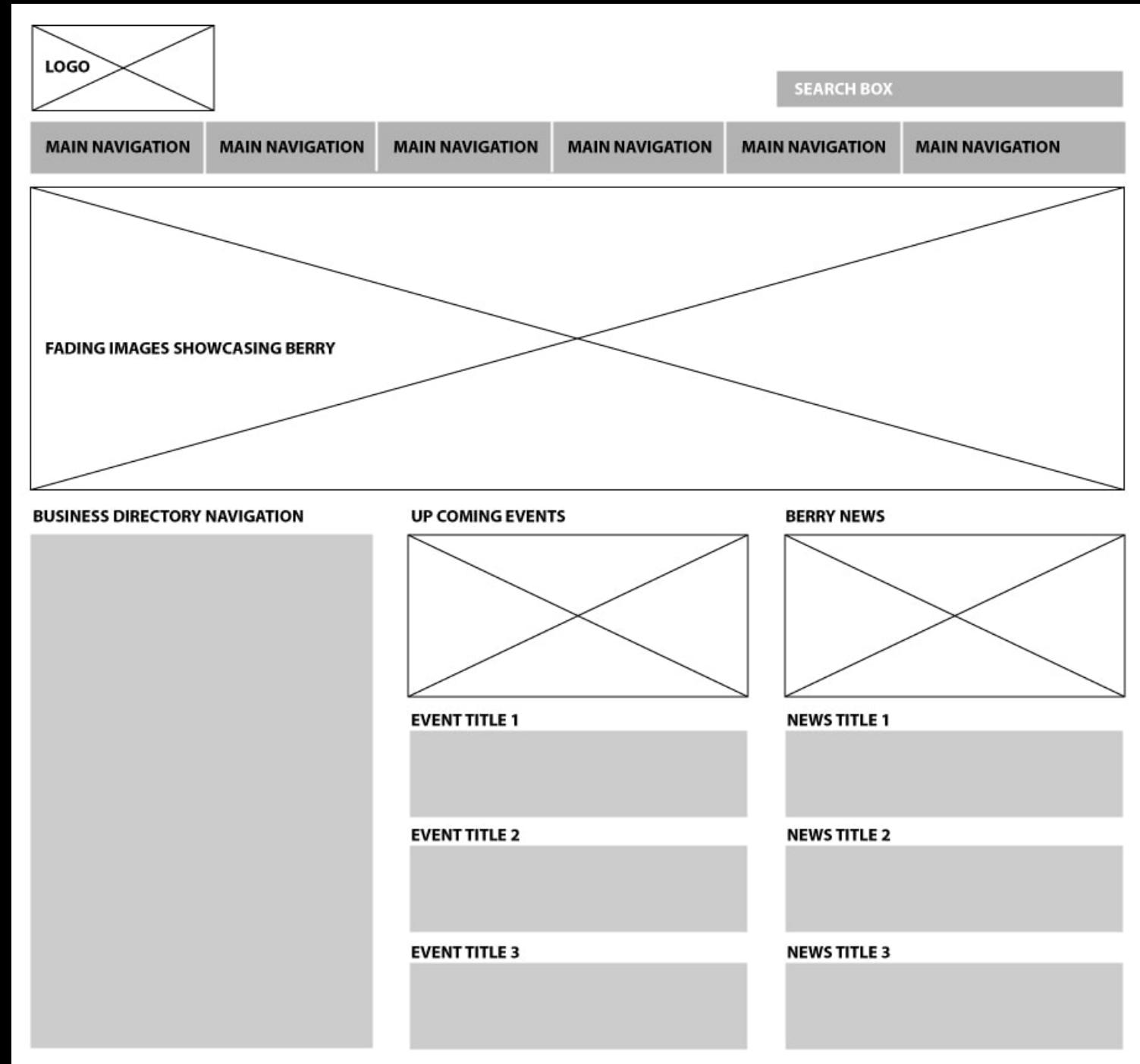
designing for the web: site maps

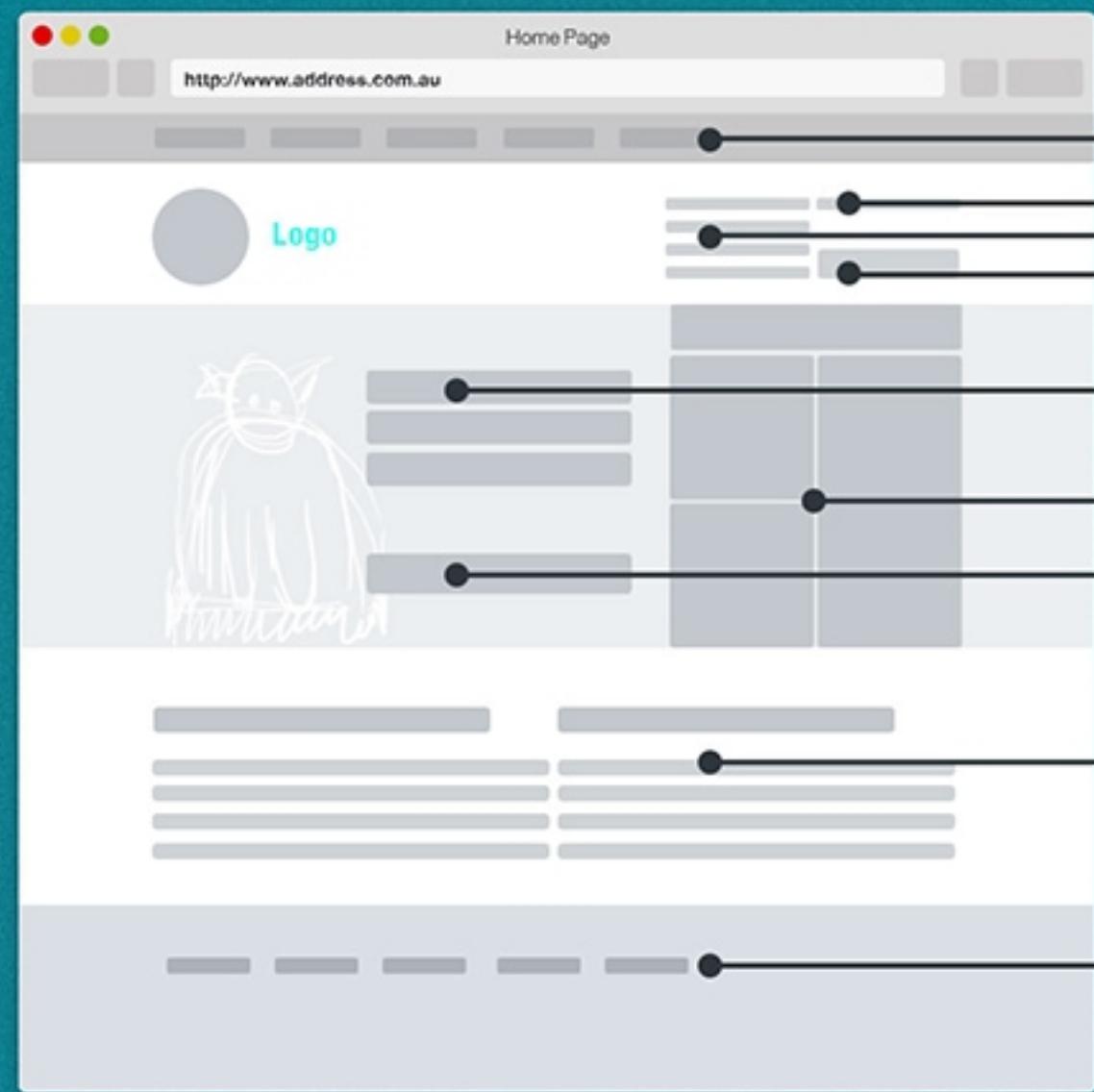




designing for the web: wire frames

Website Wire Frame





Navigation

Main navigation.

Quick Contact / Social Media

Phone/Address /Links to Social media accounts.

After Hours Highlight

Make people aware of your extended hours service.

Friendly Expertise Message

Reassuring people of your experience / care.

Links to Main Service

Graphical illustration / links to more content.

Quality / Take Action

Link, follow up on skills / expertise in the about us section

Two Highlight Boxes

Highlight pet care and expertise here, would be latest articles added to the news section

Footer Links

Contact / address / deeper linkage into website

Metadata: `viewport`

The user's visible area of a web page

HTML5 introduced a method to let web designers take control over the viewport, through the **<meta>** tag.

<!

- - Tells the browser to match the device's width for the viewport
- Sets an initial zoom value -->

<meta name="viewport" content="width=device-width, initial-scale=1.0">

CSS works by associating rules with HTML elements. These rules govern how the content of specified elements should be displayed.

A CSS rule contains two parts: a selector and a declaration.

It takes 5% to learn how to write CSS rule and 95% to learn different properties that you can use.

What does Cascading mean?

CSS Cascading has to do with how the styles apply when you have two or more rules that apply to the same element. Which one takes precedence?

CSS Rules will cascade towards specificity and then downwards on the page. For example, if you set text to be a certain size in your **<body>**, that rule will cascade all the way down to all of the elements within body

This all comes back to the tree, parent/child structure of HTML...

But: if you do something to a **<body>**, then do something to **<lists>** within the body, the latter will take precedence

If you do two things to those lists, the last one on your CSS page will take precedence

Border

Box Model

All boxes have borders even if invisible or 0px wide. It separates the edge of one box from another.

Padding

Padding is the space btw the border + any content contained within it. More padding increases the readability of its contents.



Margin

Margins sit outside the edge of the border. You can set the width to create a gap btw borders of adjacent boxes.

Content

* Why Use External Style Sheet:

Same CSS for each HTML page

No need to copy or rewrite styling

Make changes to CSS automatically + can apply to the entire website

Faster download time for subsequent pages

Selector is a term such as **p**, **h1**, **div** that identifies the HTML element you want to format or apply a rule to. You can add multiple selectors in a declaration.

Selector	Meaning	Example
Universal Selector	Applies to all elements in the document	* {}
Type Selector	Matches element names	h1, h2, h3 {}
Class Selector	Matches an element whose class attribute has a value that matches the one specified after the period (or full stop) symbol	.note {} targets any element whose class attribute has a value of "note"} p.note {} targets only <p> elements whose class attribute has a value of "note"
ID Selector	Matches an element whose id attribute has a value that matches then specified after the # symbol	#introduction {} targets the element whose id attribute has value of "introduction"

Selector

Meaning

Example

Child Selector

Matches an element that is a direct child of another

`li > a {}`

targets any `<a>` elements that are children of an `` element (but not other `<a>` elements in the page).

Descendant Selector

Matches an element that is a descendent of another specified element (not just a direct child of that element)

`p a {}`

targets any `<a>` elements that sit inside a `<p>` element, even if there are other elements nested btw them

Selector

Meaning

Example

Adjacent Sibling Selector

Matches an element that is the next sibling of another

`h1+p {}`
targets the first `<p>` element after any `<h1>` element (but not other `<p>` elements)

General Sibling Selector

Matches an element that is a sibling of another, although it does not have to be the directly preceding element

`h1~p {}`
if you have two `<p>` elements that are siblings of an `<h1>` element, this rule would apply to both

```
/* type/element selector */
```

```
p {
```

```
    color: blue;
```

```
}
```

```
/* class attribute selector */
```

```
.blue-text {
```

```
    color: blue;
```

```
}
```

```
/* id attribute selector */
```

```
#blue-par {
```

```
    color: blue;
```

```
}
```

```
/* BONUS: grouping
```

```
selector */
```

```
p,
```

```
.blue-text,
```

```
#blue-par {
```

```
    color: blue;
```

```
}
```

selecting multiple elements:

```
h1, h2, h3 {  
  
    color: red;  
    background-color: blue;  
    width: 500px;  
  
}  
  
p,  
li {  
  
    background-color: red;  
    font-color: blue;  
  
}
```

HTML comments are written like this

```
<!-- This is a comment -->
```

CSS comments are written like this

```
/* This is a comment */
```

```
{  
text-align:  
  
    left ;  
    right ;  
    center ;  
    justify ;  
  
}
```

```
{  
vertical-align:  
  
    baseline ;  
    sub ;  
    super ;  
    top ;  
    text-top ;  
    middle ;  
    bottom ;  
    text-bottom ;  
  
}
```

This property is NOT intended to allow you to vertically align text in the middle of a block level elements such as `<p>` + `<div>`, although it does have this effect when used with table cells `<td>` + `<th>` elements.

It is more commonly used w/ inline elements such as ``, `` or ``. When used with these elements, it performs a task very similar to the HTML align attribute used on the `` element.

Interaction Design

a: link {

a: visited {

: hover { Applied when a user hovers over an element w/ a mouse. This changes the appearance of links and buttons when a user places their cursor over them. Does not work on mobile.

: active { Applied when an element is bingo activated by a user, like when a button is pressed or a link clicked. This added to UX.
Applied when an element has focus. Any thing you can interact with.

: focus {

}

Focus occurs when a browser discovers that you are ready to interact w/ an element. For example when yr cursor is in an input - that element is said mohave focus.