

## MDIA 2003 Project 1 | Assignment LAB-02A

# HTML Structure

**Due date** Bring to class next week & post your GitHub or GitLab link to the assignment folder 1 day (24 hours) before class.

**Deliverables & Hand-in Requirements**

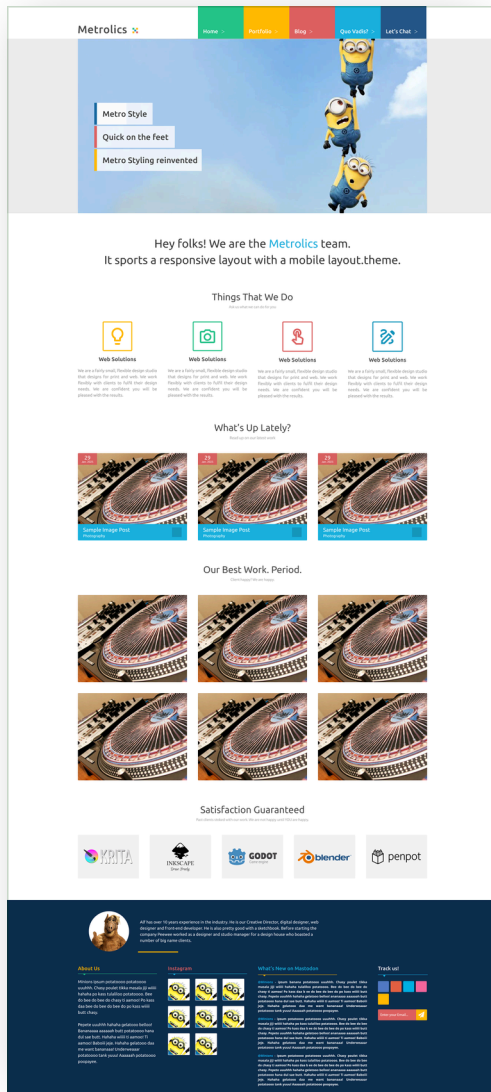
- A – HTML project based on a good foundation (consider the HTML5Boilerplate as a base)
- B – Create your HTML structure in index.html
- C – Project must be Git tracked
- D – GitHub or GitLab repository  
Bring your file(s) to class next time.
- E – This is an individual assignment for

**Description** Your assignment is to define the semantic html structure and content tagging for the provided example web page mockup. **Only the sections outlined on the next page should be added to your html code. Any CSS or graphics should not be worked on or included yet**

Aside from the base html tags, you must decide on class selectors to attach to the html structure.

**Assets** Access the Figma page mockup here:  
<https://www.figma.com/design/GAfh0fUCtk3D6gmiozuSpQ/Metrolics-Mockup?node-id=2-2&t=pvBmlE3GQZqcp0Wh-1>

## Project Brief HTML Requirements



- **include all major layout elements.**
- Think semantically.
- Consider DRY<sup>1</sup> (Don't Repeat Yourself), KISS (Keep It Simple Stupid!), OOCSS<sup>2</sup>, and BEM<sup>3</sup>.
- Use HTML5 tags or plain DIV tags, or a combination. Up to you!
- **Include the content structure:** headings, text paragraphs, image tags, links, or details (such as the area below each blog entry: categories, comments, read more link).
- **but you must NOT include or create any imagery yet!** For image tags a simple placeholder graphic will suffice. (Use the one provided.) Icons should not be added yet either.
- **one or more classes MUST be defined** and assigned for ALL tags, even if semantic HTML5 tags are used!

### Selectors Requirements

Attach class selector(s) to all tags. Consider BEM grounded methods (but feel free to come up with your own naming conventions).

### Read up on BEM

<https://getbem.com/introduction/>

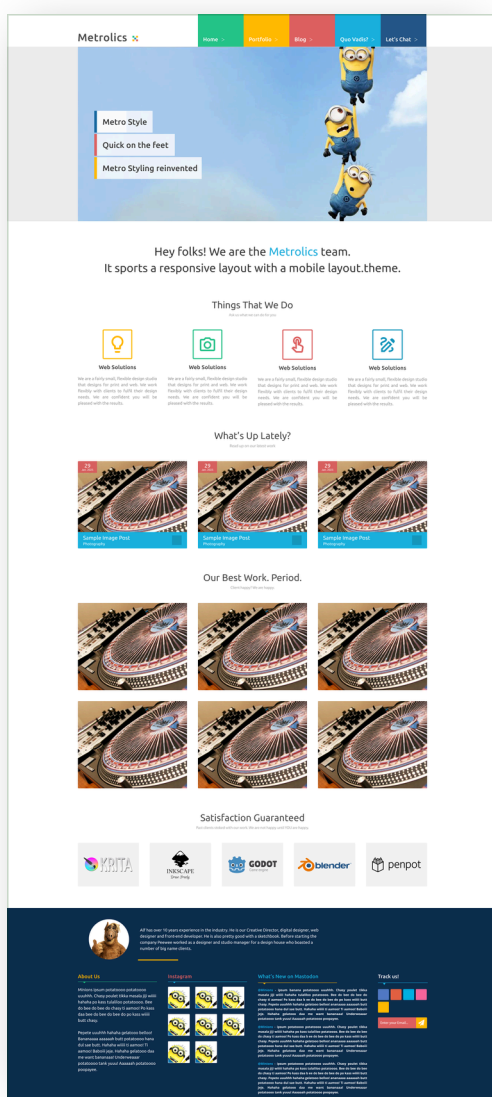
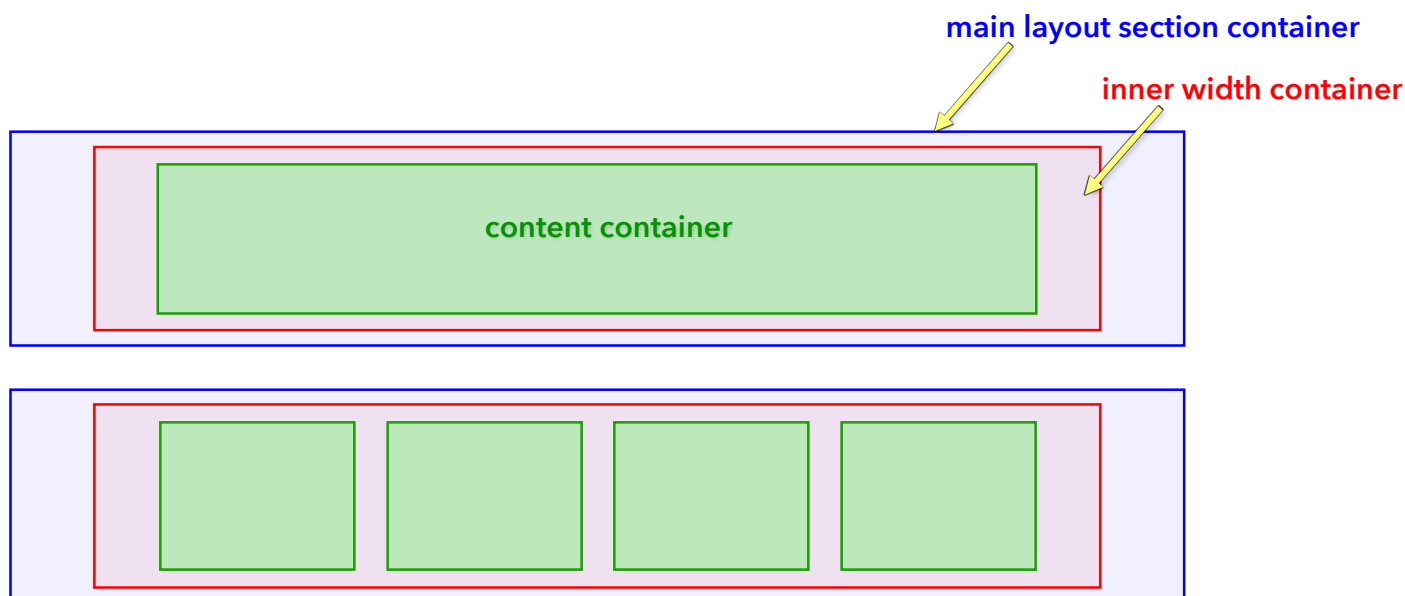
[https://sparkbox.com/foundry/bem\\_by\\_example](https://sparkbox.com/foundry/bem_by_example)

<sup>1</sup> <https://vanseodesign.com/css/dry-principles/>

<sup>2</sup> <https://github.com/stubbornella/oocss/wiki>

<sup>3</sup> <https://css-tricks.com/bem-101/>

# HTML Structure



## Hand-in Requirements

- base project folder with HTML EDITED ONLY (don't start work on CSS or JS yet!)
- index.html with base html structure code
- GitHub or GitLab hosted project
- Post remote repository information (link) to Learning Hub 24 hours before next class.

Bring your code with you next class.

The main layout HTML structure can be based on the 3-in-1 approach, which provides a simple yet flexible tag structure that lends itself well to CSS layout styling through either flexbox or CSS grid (or even the deprecated floating boxes approach).

## Project Brief Git tracking

You must initialize git in your project before you begin coding.

- Do not forget about the .gitignore file and use the provided .gitignore template.
- Track changes logically.

As a minimum example you should create commits for:

- 1] meta and language page info;
- 2] major HTML layout structure (could be one or multiple commits);
- 3] assign classes to all tags (one or multiple commits);
- 4] defining the heading structure (h1 down to h6, if necessary).
- 5] commits for inserting/defining the content parts

**It is not required to work in branches yet. Work in the main branch unless you prefer branches and know how to merge those.**

### Remote repository: GitHub or GitLab

Your local repository must be hosted on a remote repository.

If you are unsure how this is achieved, refer to the tutorial on Learning Hub in week 02.

# Rubric

<b>HTML</b>	Main layout structure tags defined	15
	Heading structure defined for all content (h1 – h6)	10
	1 or more classes defined for all tags (avoid tags without assigned classes)	20
	content structure defined	15
	Title tag, language attribute, and misc meta tags assigned values	10
<b>Git</b>	Github or Gitlab remote repository link	10
	Logical commit history: keep track of edits. Minimum requirements example commits: meta info, html structure tags, classes, headings, content parts, or similar logical managed commit history.	20
	Bonus	?
	<b>Grand total assignment</b>	<b>100</b>