



Assessment 5: Mini Project (Your creative space) (15 points)

Due: see course outline for a specific date

In this project, you will practice how to build a webpage or visual product from scratch and learn steps that will help to turn an idea to an actual webpage or a visual product. You can start working on your project from week 3 to week 5. Build your project or visual product part by part. You can start by building a top part (header) than a bottom part (footer). Then divide a main/body part into small parts and then work on each one until week 5. Anyway, the project should not exceed one page. Feel free to add more pages, but only the main page will be graded. There are three options to deliver your project, and you can choose one of them:

First options: This is an open-end project. Simply, you come up with your idea and build it. However, you must make sure the final product meets this assessment requirements and guidelines.

Second option: This option allowed you to pick a topic but suggest a page design. The design would be a vertical timeline layout (Check the suggesting layout section). You will build a timeline about any topic, e.g. iPhone development history or Windows OS releases. The timeline should not be more than ten years, weeks, or days (10 sections). It is up to you to design the timeline and structure it, or you can go with the suggested layout (Check the suggesting layout section).

If you choose to go with the first and second options, you need to make sure your topic follow these guidelines:

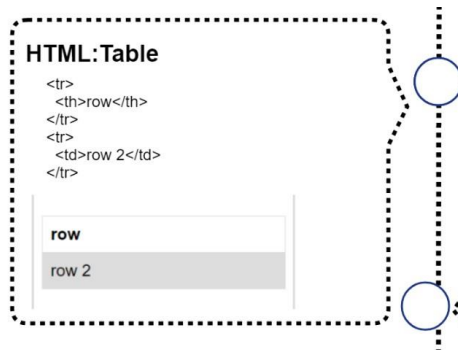
1. The final project should not include any personal information about you such as phone number, emails or social media accounts.
2. The content should not present any offensive material related to race, religion, politics, or any material that may consider publicly sensitive. In case you are not sure about your topic, feel free to contact me.
3. Your final product will be uploaded to D2L for one year, so make sure that your project materials do not have any copyright issues. You can get permission from a martial creator. Anyway, you can find free resources using keywords such as 'royalty free', 'free license' replace dots with images, audio, ... etc. Feel free to check *the free resource section* at the end of this document.
4. *** Please do not think too much about the content of the page "Don't follow in love with the content". Instead, try to think of a product that will allow you to implement html and css skills, that have been covered in this course.

Third option: You build a timeline about html and css topics. The design will contain topics about HTML tags and CSS properties such as:

- | | | |
|-------------------|--------------------|---------------|
| 5. Main html tags | 9. Text formatting | 12. Audio |
| 6. Images | 10. Video | 13. Box model |
| 7. Style list | 11. Forms | 14. CSS units |
| 8. Tables | | |

The timeline layout has three sections that are navigation menu, the timeline points, and reference.

1. Navigation menu section: Contain links point to each topic in the timeline.
2. Timeline section: It's the main or body content of the page. The content will be designed and added into cards style. A card will represent a topic. The card design can have parts or elements such as header, body/content, footer. The header contains the title of the card, body/content contains a code representing an html or css topic (e.g. table), and the footer contains the body/content code result (the result of the table tag). Here is an example:



3. Footer section: Containing reference links to extra html and css sources.

**** Check the suggested layout section to get an idea.

Project design and development plan (optional):

1. Week 3:
 - a. Brainstorm and build an idea or find a topic. If you need help, check the attached file called "Web design brief document.pdf". I would recommend that you start by writing everything come to your mind on a list. Then review the list and only keep things that essential to deliver your idea.
 - b. Sketch and collect content materials, e.g. images, media, icons... etc. Start by sketching your idea on paper before start designing. I would recommend that you start with basic or general sketching structure. Keep on your mind to sketch a structure based on the materials you have collected. So, the content would fit into the structure well. I would advice you, never start coding without having a defined design of your idea. Once you finalized your sketching, you can design it in detail using HTML and CSS.
2. Week 4:
 - a. Start coding your layout for the main screen size you picked. Start with header and footer, then work on the body content.
 - b. Code the body content.
 - c. Check your code from errors and validate it
3. Week 5:
 - a. Make your layout responsive to fit the second screen size you picked.
 - b. Check on errors, and validate your code. Then wrap up your project 🎉.

✓ Requirements:

1. The design should be organized, and your code is clear. Crumble, or broken code will be rewarded with 0. The broken code is a code that does not show a visual result or output.
2. The final product must have at least five sections (tag blocks) including header and footer.
3. At least one image, audio, and video included in the final project.
4. Fonts must be specified. You can use any fonts, but you must show it in your css code.
5. The website looks fine on two size screens. Here is the list you can choose from:
 - a. Mobile: minimum width is 600px.
 - b. Tables: minimum width is 768px.
 - c. Desktop: minimum width is 992px or 1200px.
6. Extra use of code in your project should not exceed 10% in total. That can be count by lines. If your project is 100 lines, only ten lines of extra code allowed. Anyway, that include the use of external style sheet but not the css file itself. So, if you include an external css file in your project, the only code you use from the file will be counted; but not the css code/file itself.
7. Your code should work and look fine on Chrome and Firefox browsers.

✓ Instructions (Optional but recommended)

○ Step one: Set up a project folder:

It is a good idea to create a folder for this project. The HTML document “.html” should be in the root folder (the main project folder). For extra resources such as images, create a subfolder called “images”, and for the style sheet files called it “css” or “style_sheet” ...etc. Your final project folder tree can look like this:



○ Step two: create a page layout using HTML:

Now, you need to figure out how to structure the HTML page layout. Build the layout besides on your design. Best way to start a layout is:

- First, write the main tags. Also separate your code using comments when creating a section, block or item/elements.
- Second, work in each section or item/element (header, middle, footer). Look closely at each element structure.

○ **Step three: style and format HTML layout:**

Now, you need to use CSS to style your HTML layout. Go back to your design details and translate them to css code. Consider css properties and their values for each html elements.

○ **Step four: check code errors and validate your code:**

Now, you need to check errors on your code.

- First, check syntax errors, if there is any, fix them.
- Second, check run-time errors, if there is any, fix them.
- Third, look if you have achieved the same result/output of this project, if not. Identify where is the problem for example, if the layout is not on the same order as required, check your HTML code first, then check your CSS code especially the position and display properties. If you have tried everything and still not getting a good result, please feel free to [email](#), and I will be happy to help 😊.
- Validate your html code first, then Validate your css code. If you be able to get the validation icons, please include them on your final html product.

○ **Step Five: Wrap up your project:**

Now, check all project folders and files, make sure they all open fine. Next, zip your main/root folder and only send that zip folder. Here are some links may help with file formatting:

- How to zip a file in Windows 10: <https://www.laptopmag.com/articles/how-to-zip-files-windows-10>
- Free file compressor application for Windows: <http://www.7-zip.org/download.html>
- How to zip a file in Mac: <https://www.lifewire.com/how-to-zip-and-unzip-files-and-folders-on-amac-2260188>
- Free file compressor application for Mac: <https://theunarchiver.com/>
- How to save a pdf in MS Word document: <https://www.bettercloud.com/monitor/theacademy/save-word-doc-pdf/>

Please refer to the rubric at the end of this document for evaluation details.

Submit

1. Name your folder yourname_project.
 2. Right-click on the folder and choose 'Send to -> Compressed (zipped) folder'.
 3. Upload the zipped folder to the One-page project Dropbox
-

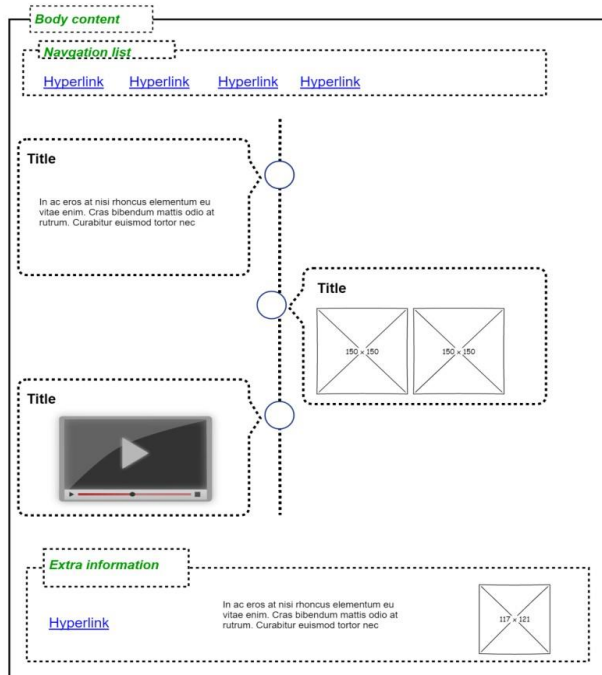
Evaluation

This assessment is graded out of 15 points and will be evaluated using the following rubric.

Learners may receive partial scores or a zero for unacceptable work.

Marking Component	Out of	
HTML layout is structured and organized	5	
The code compiles with no errors	5	
Code is validated (html and css icons are included in the page)	5	
Page style with an external style sheet.	2	
Code comments are used and clear	5	
Media (video and audio) are included	5	
Images are included	2	
Responsive:		
Page layout are responsive for two screen sizes	5	
Image resize right	3	
Font resize right	3	
	Total	40
	Final	15/15

✓ Layout suggestion section:



✓ List of software section:

○ Sketching:

- ✓ Pencil (simple): <https://pencil.evolus.vn/>
- ✓ Gimp: <https://pencil.evolus.vn/>
- ✓ MS Office
- ✓ Google Draw (Online)

○ Free text and code editor:

- ✓ Brackets: <http://brackets.io/>
- ✓ OpenOffice: <http://www.openoffice.org/>
- ✓ Google document (Online)

○ Online file management:

- ✓ Google drive: <https://www.google.ca/drive/>
- ✓ One drive: <https://onedrive.live.com/about/en-ca/>
- ✓ Dropbox: <https://www.dropbox.com/>

✓ Free resources section:

○ Images:

- ✓ Unsplash: <https://unsplash.com/>
- ✓ Gratisography: <https://gratisography.com/>
- ✓ Pixabay: <https://pixabay.com/>
- ✓ Lifeofpix: <https://www.lifeofpix.com/>

○ Videos:

- ✓ Lifeofvids: <https://lifeofvids.com/>
- ✓ Pexels: <https://videos.pexels.com/>

○ Audio:

- ✓ Videvo: <https://www.videvo.net/royalty-free-music/>

○ Icons

- ✓ icons8 (web friendly): <https://icons8.com/>
- ✓ Font awesome (web friendly): <https://fontawesome.com/icons?from=io>
- ✓ Flaticons: <https://flaticons.net/>

○ Fake text:

- ✓ Lipsum: <https://www.lipsum.com/>

○ Fonts:

- ✓ Google fonts: <https://fonts.google.com/>

- ✓ Tips: Tags cheat sheet. Feel free to use any tag that is not listed below:

Head	Sections	Text-level	Grouping	Forms	Embedded
1. doctype 2. html 3. head 4. title 5. base 6. link 7. meta 8. style 9. script 10. noscript	1. body 2. article 3. nav 4. aside 5. section 6. header 7. footer 8. h1-h6 9. main 10. address	1. a 2. em 3. strong 4. i, b 5. u 6. s 7. small 8. abbr 9. q 10. cite 11. dfn 12. sub, sup 13. time 14. code 15. kbd 16. samp 17. var 18. mark 19. bdi 20. bdo 21. ruby, rt, rp 22. span 23. br 24. wbr	1. p 2. hr 3. pre 4. blockquote 5. ol 6. ul 7. li 8. dl, dt, dd 9. figure 10. figcaption 11. div	1. form 2. fieldset 3. legend 4. label 5. input 6. button 7. select 8. datalist 9. optgroup 10. option 11. Textarea 12. keygen 13. output 14. progress 15. meter	1. img 2. iframe 3. embed 4. object 5. param 6. video 7. audio 8. source 9. canvas 10. track 11. map 12. area
Interactive	Tables		Edits		
1. details 2. summary 3. command 4. menu	1. table 2. caption 3. thead 4. tbody 5. tfoot 6. tr 7. th 8. td 9. col 10. colgroup		del, ins		

To learn more about any tag, search W3C website.

<https://www.w3schools.com/tags/>

To learn more about any CSS property, search W3C website:

<https://www.w3schools.com/cssref/>