



**Swinburne University of Technology**  
**Swinburne College**  
**TP1/2023 – COS10024 Web Development**  
**Assignment 1**  
(Develop a simple static website)

**Important Dates:**

<b>Due Date Canvas</b>	<b>End of Week 6 – See Canvas (On or before 9<sup>th</sup> April 2023, 23:59)</b> (Late submission penalty 10% of total available marks per day)
<b>Demonstration</b>	Your tutorial session 1: Week 7

**Contribution to Final Assessment: 15%**

**Note:** You must meet the Essential Requirements of this assignment to be eligible to submit Assignment 2.

**Purpose of the assignment:**

An individual assignment to familiarise students with the techniques and skills involved in creating webpages using validated HTML5 and validated CSS; loading and testing the website on the server; and demonstrating an understanding of HTML5 elements and attributes, CSS selectors and properties and their proper use. Moreover, it is to provide students with an appreciation of the importance of the separation of structured content from presentation as discussed in the lectures.

To highlight this, the assignment is specified and evaluated in three sections.

- The development of validated HTML5 webpages.
- The development of validated CSS files to present webpages on various devices.
- The implementation of CSS animations.

**This is an Individual Assignment. *All work must be your own.* Submissions are *automatically checked* for similarities. Unexplained similarities may constitute plagiarism. Carefully read the section on **plagiarism** in the Unit Outline before you proceed (including the section forbidding sharing your work with others).**

**Note:** *No JavaScript (JS) is to be used in this part of the assignment.*

**Section One: HTML5 (40 Marks)**

Ceylon Textile Suppliers (CTS), an Australian owned organization based in Melbourne which supplies clothing related products to Australian and New Zealand retail outlets on a sale or return basis. It has an annual turnover of around \$ 60M and plans to expand. CTC wants to develop a website that will enable it to advertise vacant positions. These have a 'position description' that sets out the qualifications, skills and knowledge required. Potential applicants for the position will be able to submit an online form to apply for a position.

Create at least **Five** web pages using HTML5 and with at least **one image on each web pages, three images in total minimum**. All images must be original, meaning you own those images.

The Information content on these webpages should be information about you. Please make sure to include:

- Homepage with details of the company (**index.html**)
- A page of job descriptions (**jobs.html**)
- A job application page (**apply.html**)
- A page with your personal details (**about.html**)
- A page which lists any enhancements not know they exist!  
you have made (**enhancements.html**)

You must call these files **exactly** by these names, otherwise the Tutor will not know they exist!

## Website Folder Structure:

assign1/	<i>You must have this folder –case sensitive!</i>
index.html	
jobs.html	
apply.html	
about.html	
enhancements.html	
images/	Folder for images for your page contents
styles/	Folder for style css files
styles/images/	Folder for images referred to by your css files Eg. Background

### 1. Homepage (**index.html**)

This page should contain appropriate title, a description and graphics related to the company. It is up to you to make up the details of the company that is advertising the jobs. It should contain a menu that links to the other pages on your Web site. This same menu should be in every page of your website with an email link to your student email.

### 2. Position Descriptions page (**jobs.html**)

You need to write a web page with at least 2 position descriptions. *For one of these, your tutor will allocate you a job title from the above scenario.* For the second position, the choice of job type is entirely up to you.

The HTML on this page must contain:

- Hierarchically structured headings of at least 2 levels
- More than one <section>
- An <aside> with appropriate content
- At least one ordered list

- At least one unordered list
- The page should also have an appropriate footer.

Your job descriptions should be concise but as a minimum include:

- Company's position description reference number (5 alphanumeric characters)
- Position title
- Brief description of the position
- Salary range
- The title of the position to whom the successful applicant will report
- Key responsibilities. A list of the specific tasks that are to be performed
- Required qualifications, skills, knowledge and attributes. These should be divided into "essential" and "preferable". These requirements should include such things as specific technical knowledge, number-of-years' experience required, etc.

The content of the job description should be appropriately structured with headings, sections, subsections, lists etc. using the appropriate HTML elements.

**Sources / References:**

- You can use material from other websites, but the source of all material must be acknowledged. This acknowledgement should be immediately after the material and include a **hyperlinked URL** to the original source. The text of the hyperlink reference can be a short name, but the hyperlink must work.
- If you are unsure of what is contained in a position description, there are many resources on the internet. This is one of the examples:  
<https://recruitloop.com/blog/how-to-write-a-job-description/>

### 3. Job application page (*apply.html*)

This page has a form that allows a potential candidate to register their interest in the advertised position. HTML5 data validation should be used to check the user's input.

The form will allow a potential applicant to fill in the following:

Field	Format requirement
Job reference number	exactly 5 alphanumeric characters
First name	max 20 alpha characters
Last name	max 20 alpha characters
Date of birth	dd/mm/yyyy
Gender	radio inputs grouped using a field set and legend
Street Address	max 40 characters
Suburb/town	max 40 characters
State	drop down selection from VIC, NSW, QLD, NT, WA, SA, TAS, ACT
Postcode	exactly 4 digits
Email address	validate format

Phone number	8 to 12 digits, or spaces
Skill list - the last item in list should read "Other skills..."	checkbox inputs
Other skills	Text area

All inputs should have labels. All form values, except the comment text area are 'required' or have a default value (e.g. select and checkbox inputs). ***The user should not be able to submit the form if any of these required fields are blank.***

#### Data Submission to Server

The form should have a submit button labelled "Apply". When this button is clicked the name- values from the associated form should be sent to the server using the post http method. The server action address is <https://mercury.swin.edu.au/it000000/formtest.php>. The server will then just echo back the name-value pairs to the client. While nothing will be stored on the server in this part of the assignment (we will do this in Part 3) this will allow the form submission to be tested.

#### 4. A page about you (*about.html*)

This page will contain information on the following:

Information	HTML element to be used
Your name	Definition list
Student number	Definition list
Your tutor's name	Definition list
Course you are doing	Definition list
Photo of you < 100k	HTML figure element
Your Swinburne timetable	HTML table
A <b>mailto</b> link to your student email.	

It could also include personal profile, such as resume, interests, or information that is related to you. This extra information gives you an opportunity to extend the techniques you apply in your assignment, and could include:

- Demographic information about you
- Description of hometown
- A list of your favourite books, music, films etc.

**[IMPORTANT]** All webpages must be fully validated against HTML5 without errors, on <http://validator.w3.org> or using Web Developer (Firefox add-on or Chrome extension).

The following HTML elements **MUST** be used in this assignment. Some will be used on all pages while others might only be used once.

- General elements `<head>`, `<title>`, `<meta>`, `<body>`
- Structural elements `<header>`, `<nav>`, `<main>`, `<footer>`, `<aside>` o `<article>`, `<section>`, `<aside>`
- Content elements `<h#>`, `<p>`, `<hr />`  
`<ul>`, `<ol>`, `<table>`, `<img>` and `<a>`
- Form elements `<form>`, `<input>` and `<label>`

**[IMPORTANT]** The use of the form must fit the context of the web page.

Form data should be submitted to:

<http://mercury.swin.edu.au/it000000/cos10005/surveytest.php>

Other HTML elements such as `<div>`, `<span>`, `<blockquote>`, `<strong>`, `<em>`, `<fieldset>` and more can be used as necessary and appropriate for the content.

A menu must be created on each web page that allows users to navigate through all the webpages on your website. This menu must be properly styled using CSS.

Please organise your HTML code properly and annotate the source code with comments as necessary.

## CSS Requirements [40 marks]

Develop **TWO** CSS files that style the web pages, one for a **desktop** and another for a **mobile phone**. The webpages must be styled to suit the corresponding device, e.g., the layout, font size and the sizes of the major components of the webpages.

Demonstrate a range of selectors, including but not limited to:

- |   |            |       |   |
|---|------------|-------|---|
| ? | element    | e.g., | <code>h1 {}</code> , <code>p {}</code>                |
| ? | class,     | e.g., | <code>.image {}</code> , <code>.highlighted {}</code> |
| ? | id         | e.g., | <code>#tbName {}</code>                               |
| ? | grouping   | e.g., | <code>h1, h2, h3 {}</code>                            |
| ? | contextual | e.g., | <code>ul li {}</code> pseudo-                         |
| ? | classes    | e.g., | <code>:hover {}</code>                                |

Demonstrate the following groups of CSS properties:

- Menu *e.g., style a navigator that contains links to all web pages*
- Web Typography *e.g., fonts, colours, text alignment*
- Box Model and Visual Formatting *e.g., border, margin, padding, width, height, background image*
- Liquid Page Layout *i.e., page layout adapts as you resize the browser window*

Use of other CSS selectors and CSS properties are encouraged, as necessary and appropriate for styling the webpages.

Please properly organise your CSS code and include comments within your CSS code as needed. e.g., /\* banner style \*/.

Creativity is strongly encouraged in the web site design. Creative designs will be rewarded with bonus marks.

**[IMPORTANT]** All CSS files must be fully validated against CSS3 without errors, on <http://validator.w3.org> or using Web Developer (Firefox add-on or Chrome extension).

## Enhancements to the Specified Requirements (CSS Animations) [20 marks]

***Note: Make sure you get all the basics working first before you attempt any enhancements. See the marking Guide below.***

The technologies for developing Web applications are rapidly changing. One of the key skills you will need is finding out about these new techniques and applying them. When researching, look at the reliable websites such as the External Links provided on Canvas. This assessment gives you an opportunity to demonstrate your ability to implement features/techniques that go beyond the specified requirements above. It also provides you with an opportunity to demonstrate your ability to discover techniques from a range of sources and apply them in a standards-compliant manner.

Implement 2 different CSS animations. There are many types of CSS animations. Take your pick and make sure they are applied to appropriate HTML elements. Feel free to use Google or any source of information you deem appropriate to look up the code you need to implement the CSS animations. Create a fifth page named ***enhancement.html*** and incorporate it into the menu navigator. On this webpage, list the animations that you have implemented and provides hyperlinks to where the animations are implemented.

## Assignment Submission

The assignment should be submitted as an individual work through CANVAS (**Electronic Submission**) and into the **mercury sever** on or before the deadline.

- Create a folder/directory “assign1” for your website files and folders. In folder “assign1”, create a subfolder “images” for images used for content and a sub-folder “style” for your CSS files.
- Compress folder “assign1” and all your HTML files, images files and CSS files into a zip file named “assign1.zip” and submit it through CANVAS. • All deliverables must be submitted in this zip file.

**Before submitting** the assignment, please ensure that you have completed or considered the following:

- Check if you have fulfilled the requirements and avoided the deductions in the Marking Scheme
- Check the Canvas for announcements related to the assignment.
- The assignment is your original work. If your assignment is not your original work, your result will be zero marks for this assignment.

- Validate and test your webpages on the Mercury server. If your assignment cannot be fully validated or properly displayed on the Mercury server, penalties will apply.
- Ensure that you submit before the due time to avoid penalties.
- Make sure you contact the convenor before the due date, if you are having trouble in completing the assignment on time.

### Marking Scheme

The assignment will be marked out of 100.

	Mark	Requirements	Deductions
<b>HTML5</b>	40	<ul style="list-style-type: none"> <li>• All required HTML elements created (10)</li> <li>• At least 3 HTML pages of original content (5)</li> <li>• Webpages fully validated against HTML5 &amp; Quality of your web pages (10)</li> <li>• Source code properly indented, commented and organised (5)</li> <li>• Original photos used (5)</li> <li>• At least 3 images in appropriate sizes (5)</li> </ul>	<ul style="list-style-type: none"> <li>• Web pages not fully validated against HTML5</li> <li>• Web pages not displayed properly</li> <li>• Not enough web pages</li> <li>• Lack of required HTML elements</li> <li>• Lack of required contents</li> <li>• Deprecated HTML elements/attributes used</li> <li>• Embedded or inline CSS</li> <li>• Links on web pages not working</li> <li>• Unoriginal contents</li> <li>• Lack of required images</li> <li>• Unoriginal images</li> <li>• Images too large or too small</li> </ul>

<b>Basic CSS</b>	40	<ul style="list-style-type: none"> <li>• All major components of web pages properly styled, e.g., <code>&lt;header&gt;</code>, <code>&lt;main&gt;</code>, <code>&lt;footer&gt;</code> and <code>&lt;aside&gt;</code> (10)</li> <li>• All required CSS selectors demonstrated (3)</li> <li>• All required property groups demonstrated. (5)</li> <li>• 2 CSS files, one for desktop and the other for mobile phones. (5)</li> <li>• Appropriate and fluid page layout. (5)</li> <li>• CSS files fully validated against CSS3 (5)</li> <li>• CSS code properly indented and organised (3)</li> <li>• CSS properties logically grouped (2)</li> <li>• Consistent styles for all web pages (2)</li> </ul>	<ul style="list-style-type: none"> <li>• CSS files not fully validated against CSS3</li> <li>• CSS files not properly applied to web pages</li> <li>• Lack of required CSS file(s)</li> <li>• Not all major components of web pages properly styled</li> <li>• Inappropriate web page layout</li> <li>• Inappropriate contrast between foreground and background text colour</li> <li>• No major difference in page layouts for desktop and mobile phones</li> <li>• No comments or unclear comments</li> <li>• Lack of required CSS selectors</li> <li>• Inconsistent styles for different web pages</li> </ul>
<b>CSS Animations</b>	20	<ul style="list-style-type: none"> <li>• 2 different CSS animations implemented. (10*2)</li> </ul>	<ul style="list-style-type: none"> <li>• CSS animations do not work properly.</li> <li>• Brief introduction about your animation</li> <li>• References related to your animation</li> </ul>
<b>Total</b>	100		