

COS10026 Web Technology Project

Applied Web Project Part 1 of 2,

Semester 1, 2025

Due Date	11:59 pm on Monday in Week 7 (April 14th) (Late submission penalty: 10% of total available marks per calendar day)
Contribution to Final Assessment:	40% (70% group tasks , 30% Individual tasks)

Important: Submissions are *automatically checked* for similarities. Unexplained/acknowledge 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). The Group Agreement must have been uploaded before this assignment.

Project Part 1 of 2 (Group Tasks)

Purpose

This assignment will familiarise you with the techniques and skills involved in designing and creating static webpages utilising validated HTML and CSS created with a standard text editor. You also learn about accessibility guidelines for web in this assessment. You will deploy these Web pages on a Unix / Apache server. This should be done in a way that keeps HTML content and CSS presentation separate, as discussed in the lectures.

No JavaScript is to be used in this part of the assignment.

Essential Requirements

Scenario:

An **IT company** 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.

In this assignment you will develop a prototype of this website. The website you develop will consist of the following Web pages, accessible from a common menu on each page:

- Home page with details of the company ([index.html](#))
- A page of job descriptions ([jobs.html](#))
- A job application page ([apply.html](#))
- A page with your group details ([about.html](#))

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

You will also include

- A CSS file that styles your website ([styles.css](#)).

The **essential requirements** for this assignment are listed in the marking guide. In general the web pages must:

- have relevant content
- must include the HTML markup specified in the marking guide
- must validate to HTML5 without errors
- must be styled by a validated CSS3 file
- must be linked to each other via a menu
- must be deployed on a GitHub Repository.

All web pages in your website should have a consistent layout and navigation. Where “in-house” templates have been defined in this unit (e.g. for meta-data; tables; etc.) these should be followed. These include accessibility alternatives.

The HTML in your Web pages must validate against the W3C HTML5 validator (<http://validator.w3.org/nu>) .

Pages should not contain any deprecated elements/attributes (e.g. `<i>` , ``). Do not use iframe elements in your assignment.

Note: Generic structural elements like div or span should only be used where there is no more meaningful HTML5 element that is appropriate.

1. Home page (*index.html*)

This page should contain appropriate title, a description and graphic related to the company (with company name, and logo). 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 the companies email in the format of `info@companyname.com.au`

In the footer include the link to your Jira Project (Don't forget to give access to your tutor)

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

You need to write a web page with at least 2 position descriptions. **Each group will get a job title posted on Canvas**. For the second position, the choice of IT job type is entirely up to you. Why not write a position for the ideal job you would like to do? 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 and semantic tags.

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 programming languages required, number-of-years of 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 or generated by GenAI but the source of all material must be acknowledged. This acknowledgment should be immediately after the code as a comment *and* include a **hyperlinked URL** to the original source. If you are using GenAI you need to comment your prompt message.
- If you are unsure of what is contained in a position description there are many resources on the web. Here is one: <http://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	drop down selection including all jobs ref numbers
First name	max 20 alpha characters
Last name	max 20 alpha characters
Date of birth	dd/mm/yyyy
Gender	radio inputs grouped using a fieldset 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 based on States
Email address	validate format with regular expression & patterns
Phone number	8 to 12 digits, or spaces
Required technical list	checkbox inputs
Other skills	textarea

All inputs should have labels. All form values, except the "Other Skills" textarea 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 or incorrectly filled.***

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 2) this will allow the form submission to be tested.

4. A page about your group (*about.html*)

This page will contain information on the following:

Information	HTML element to be used
Your group name - Class time and day	A Nested List (Any)
All your student IDs	
Your tutor's name	
Members Contribution to this project	Definition list (Name, Contribution)
Photo of your group (Group photot) < 300k	HTML figure element
Members Interests	HTML table (Most have cell or row merged with a caption)

It could also include group profile, such as programming skills, working experiences,, or information that is related to your group. This extra information gives you an opportunity to extend the techniques you apply in your assignment, and could include:

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

5. CSS Requirements

No style markup should be included in your HTML file.

The pages in your website must be styled with CSS and have a consistent 'look and feel', particularly common elements such as menus, headers and footers. While the emphasis is this assignment is on the appropriate application of techniques rather than graphic design, your pages should follow basic usability / accessibility principles, e.g. distinguishable foreground and background colours, and font readability, etc.

Create your own design and implement it using one **single external** stylesheet that applies to *all* your Web pages. This file should be named **styles.css** and placed in a styles folder. The stylesheet should style the common elements on *all* your web pages, and address the following specific style requirements.

1. **Comments:** The CSS should include comments at the beginning of the CSS file to identify author and purpose. Individual line comments should be used as necessary to explain particular styles and explain where they are applied.
2. **Selectors:** *All* the following CSS Selectors should be used *appropriately* at some point in this assignment:
 - element, #id, .class, grouping, contextual
 - pseudo class, pseudo element
3. **Menu:** The menu should have its own set of styles applied. Use a background colour.
4. **Index Page:** Demonstrated the following specific CSS rules on the **index.html** page:
 - display a background graphic.
5. **Position Descriptions Page:** Demonstrated the following specific CSS rules on the **jobs.html** page:
 - <h1> elements should have their font variant, size and family etc. set using the short-hand **font** property.
 - The <aside> should be 25% of the width of page and float to the right.
 - The <aside> should have a coloured border with an appropriate margin and padding.
 - The footer should cover the full width of the page the footer text should be in a small font and centred in the footer.

6. **About Page:** Demonstrated the following specific CSS rules on the **about.html** page:
 - Style the student IDs on the right side of the page.
 - The photo should be styled with a single border using the short-hand **border**-property, and the figure should be floated to the right of the page.
 - <table> should be centred within the section, headings in bold, table cells with a background colour specified in hexadecimal format with some hover effect.
7. **All pages:** should have a fluid layout (the page should “Reflow” on page resize).
8. **Apply.html:** must have a display factor (e.g., flex or grid), use margin and padding, with inputs, labels and button styling.

Other CSS selectors and properties can be used as necessary and appropriate for the presentation

Do not include any proprietary CSS mark-up, such as –moz- or –webkit etc.

Hint: CSS validators will validate against a particular version of CSS e.g. CSS2.1 or 3. This assignment should be valid CSS2.1 or CSS3. Make sure that you are checking your CSS using the correct version of the validator. For example, if you include CSS3 markup and validate as CSS2.1 it will show errors. (Best to pre-set the version in the Web Developer tools – see the note on Blackboard).

All html and CSS should be commented.

6. Project Management (Jira)

Students must demonstrate effective use of Jira for project management by ensuring the following requirements are met:

- **User Stories:** Clearly defined with a proper description, assigned priority, and allocated team members. There should be at least one user story for every requirement.
- **Sprint Planning:** A minimum of two sprints must be created for weeks 5 and 6, showing a structured workflow. You can also have a sprint for week 4
- **Task Management:** The project should include epics and tasks, ensuring comprehensive project organisation.

IMPORTANT: Include the link to your Jira project as a link in the footer of index.html and don't forget to give access to your tutor.

Web Site Folder Structure and Deployment (Group Tasks)

The directory structure of your website is described below. You can create additional HTML files for your content (depending on what your content requires), but the following is needed:

project1/	<i>You must have this folder – case sensitive!</i>
index.html	
jobs.html	
apply.html	
about.html	
and other html pages	
images/	<i>Folder for images for your page content</i>
styles/	<i>Folder for styles.css other css files</i>
styles/images/	<i>Folder for images referred to by your css files e.g. background</i>

Notes:

- HTML files should only be in the base “project1/” folder – not anywhere else.
- All images used for the **content** should be stored in the “project1/images/” folder.
- All images used for the style should be stored in the “project1/styles/images/” folder.
- There should be a “styles.css” file in the “project1/styles/” folder.
- All links to your files (CSS or images) should be **relative**. **Do not use absolute links**, as these links will be broken when files are transferred for marking. No marks will be allocated if links are broken.

Note: DO NOT INCLUDE VIDEO OR OTHER LARGE (>5MB) MEDIA FILES IN YOUR SUBMISSION.

Make sure you thoroughly test your website deployment on the GitHub.

Note: Failure to acknowledge the source of third party code, GenAI or content is plagiarism and may result in zero marks for this assessment or other penalties in accord with Swinburne policy.

Deliverables

The marks are allocated 70% for group tasks (28/40) and 30% (12/40) for individual tasks in this assignment. The followings are all for group tasks only.

- Complete Website, implemented on GitHub Repository and the live link is provided in the comment section of the file submission.
- The zip file of the downloaded code from the GitHub
- Project Management done in Jira (link included in the footer of the index.html)

Project Part 1 of 2 (Individual Tasks)

Project Self Report (Template is in the assignment folder)

This self-report is an opportunity for you to reflect on your individual contributions to your static web development project. You will document your work in GitHub, your participation in team discussions on Discord, and your overall experience with the project. This report will help assess your technical contributions, collaboration skills, and learning outcomes.

Submission Requirements:

You must complete and submit a structured report covering the following sections:

1. GitHub Contribution Summary

Commit History: Provide details on your commit activity, including the branch name, total commits, commit frequency, and examples of meaningful commit messages.

Contribution Breakdown: Outline the key features and sections you worked on, the user stories you handled, and any challenges you faced, along with how you overcame them.

Code Quality and Best Practices: Describe the coding standards you followed, such as semantic HTML, proper file structure, accessibility improvements, and CSS optimisation.

2. Discord Participation

Contribution Rating: Rate your participation (1-5) and provide examples of your contributions, such as answering questions, sharing resources, or offering support.

Evidence of Participation: Indicate whether you sought or provided help, and describe an example of how you engaged with your team.

3. Final Project Assessment

Project Outcome: Reflect on whether the final project met your expectations and explain why.

Personal Contribution: Identify the strongest aspect of your contribution.

Accessibility Improvements: Describe what you did to enhance website accessibility and include supporting screenshots from the WAVE tool.

4. Reflection on Learning and Collaboration

Key Takeaways: Share what you learned from this project.

Future Improvements: Discuss what you would do differently in future projects.

Submission Format:

Submit your report as a PDF or Word document. Ensure it is well-structured and clearly formatted

Peer Review & Feedback (Individual Task)

Every student needs to complete the peer and self-evaluation. This task is crucial for providing constructive feedback to your peers and reflecting on your own contribution. The link can be found in the assessments page under "Project Part 1". The Peer & Self Evaluation, results will be reported to each member anonymously to provide an opportunity for improvement. You must provide respectful and constructive feedback.

Deliverables

The marks are allocated 70% for group tasks and 30% for individual tasks in this assignment.

- Project Self Report uploaded on Canvas individual task link (individual task)
- Fill out the Peer & Self Evaluation Form on Canvas (individual task)

Overall Assignment Submission (Canvas + GitHub)

Your website should be completely coded and developed on GitHub on or before your deadline. An electronic copy of your assignment should be submitted through Canvas on or before your deadline.

- Make sure to have regular commits in your GitHub otherwise marks will be deducted.
- Make sure all your website files are in the correct folders and compress your root folder with all your sub-folders with HTML, CSS, and images into a zip file named "groupName_part1.zip".
When you download this from the GitHub repository everything should be there. **Submit this to Canvas by your group leader (Only one person) into "Project Part 1 of 2 (Group Submission)".**
When the zip file is decompressed, the entire website should be able to be run from index.html without needing to move any files.
- You don't need to submit the Jira files. You only need to include a hyperlink in the footer of index.html page pointing to your Jira project. Ensure that your tutor has access to your project.
- You need to post the link to your live website as the submission comment, otherwise a penalty will be applied
- You need to demonstrate your work before the submission deadline to your tutor. **Failing to do so will encounter up to 50% penalty.**
- **Every student needs to do their peer and self evaluation form (penalty up to 30%).**
- Note that all deliverables must be submitted electronically and email submission is not acceptable.

Submission Summary:

- Demonstration of your website to your tutor one week before the due date.

Project Part 1 of 2(Group Submission):

- Zip file named groupName_part1.zip that includes your website code
- Jira project link on the footer of index.html page
- Website live link as a submission comment.

Project Part 1 of 2 (Individual Submission):

- Report
- Peer & Self Evaluation Form

Good Practice / In-house Standard

(It will cause deductions if these good practice / in-house standards are not followed)

Requirement
<i>Page design</i>
- Well designed structure
- Appropriate contrast in colours
- Appropriate use of fonts
- Consistent application of style across pages
- Appropriate application of styles
<i>Content</i>
- Job descriptions have sufficient quantity (200 words +)
- Job descriptions is sufficient quality
- About.html content meets spec
- Images (including portrait) present and appropriate file size
<i>HTML</i>
- Meta-data follows in-house standard
- HTML has no embedded Style markup CSS is fully separated from HTML
- No deprecated elements/attributes used
- No inappropriate use of HTML semantics (e.g. use of <div> when <section> <article> should be used)
- HTML follows usability standards (e.g. alt on images)
- Comments adequate
<i>CSS</i>
- No redundant CSS or unused selectors
- Responsive design
- Appropriate use of selectors (e.g. Class versus ID)
- Appropriate line comments
<i>Web site</i>
- Directory Structure as defined above
- Third party content inadequately acknowledged
- Accessibility Guideline is followed and implemented
Note: Failure to acknowledge the source of third party code or content is plagiarism and may result in zero marks for this assessment or other penalties in accord with Swinburne policy.