Swinburne University of Technology

School of Science, Computing and Engineering Technologies

Web Development
COS10005

Assignment 2, Semester 2, 2023

Due Date	Saturday 21th Oct 2023 at 11:59 pm
Submission Method	Canvas
Contribution to Final Assessment	40%

Purpose of the assignment: It is an individual assignment that provides an opportunity to practice again techniques and skills involved in designing webpages using HTML5 and CSS. JavaScript will be used to validate user input data before they are submitted to the server for processing. Moreover, JavaScript will be used to add more interactivity to the webpages.

Requirements and Specifications

The assignment requires that you develop a website for Sweet Life, an ice-cream shop. The director of Sweet Life, Ms Bianco, is interested in developing a website that allows users to register and place orders for ice-cream online.

Task 1: Website Design

After several meetings with the Ms Bianco, it is agreed that 3 webpages will be developed, the home page, the registration page and the order page.

1.1 Folder Structure

The directory structure shown in Figure 1 is to be created.

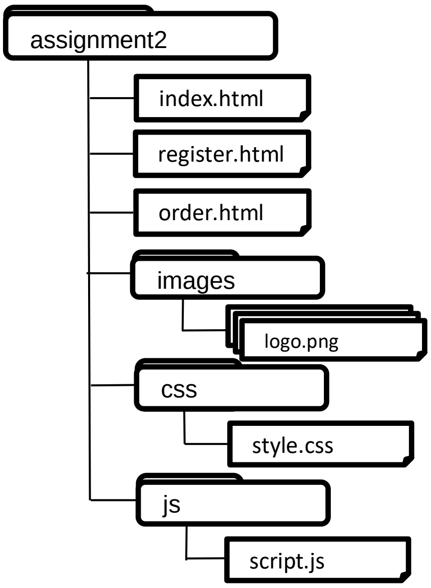

Figure 1: Directory structure

Additional files may be added depending on your own design.

1.2 Website Structure

Choose your own webpage structure. A fully functional navigator must be provided on each webpage. Users should be able to access all 3 webpages using the navigator.

1.3 Webpage Layout

You need to design an appropriate webpage layout for Sweet Life. HTML5 structural elements must be used (e.g. <header>, <nav>). You are free to use other structural elements, e.g., `<div>`.

Task 2: Website Development

2.1 HTML 3 Webpages

The three webpages must be developed using HTML5. All pages must **not contain any deprecated elements/attributes.** The content requirements for each webpage are as follows:

- Page `index.html` is the homepage of the website. It should provide introductory information about Sweet Life, its history, menu, location, available types of ice-cream, etc. Feel free to refer to real-world shops websites for example. Information about at least three types of ice-cream must be provided, with at least three original or explicitly referenced images.
- Page `registration.html` is for users to register. A form must be provided, which contains different input elements that allow users to enter registration information, e.g., username, password, gender, email, favourite ice-cream, etc. The following input elements must be used: ○ ~~label~~ ○ ~~textbox~~ ○ ~~password~~ ○ ~~radio button~~ ○ ~~checkbox~~ ○ ~~select list~~ ○ ~~submit button~~
- Page `order.html` is for users to place orders. A form must be provided to allow users to enter the following data to place an order:
  - Delivery or pickup
  - Delivery address (only if delivery is selected) ○ Billing address ○ Contact number ○ Email for receipt ○ Pay on pickup or online
  - Credit card information (only if pay online is selected)

Choose the right types of input elements for different required information in the forms, e.g., a textbox for username and a group of checkboxes for favourite icecream types.

Other input elements can be used in your forms if necessary and appropriate for the content. Set the `method` attribute of the form to `POST` and the attribute `action` to: http://mercury.swin. edu.au/it000000/cos10005/regtest.php

All webpages must be fully validated.

2.2: CSS Files

At least one CSS file is to be created to style all three HTML pages. Use any CSS properties you deem necessary and appropriate for the presentation. You are encouraged to be creative in your design. You will mark not only on the use of CSS properties and selectors but also on how professional, beautiful and creative your webpages look.

2.3: JavaScript File

A JavaScript file must be created for the registration and the order pages to validate the input data before forms are submitted to the server for processing. The JavaScript must

- check that none of the required input fields are blank
- check that the password is at least 9-character long
- check that the postcode is 4-digit
- show an alert or inline error messages to display all the errors in the input fields
- allow the form to be submitted only if all the input data are valid

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.

Task 3: Accessibility

It is necessary that you design your web pages considering accessibility rules (refer to the rubric for more details). You must submit the screenshots of WAVE’s accessibility checker to Canvas.

Implement the following to make your website accessible:

- Each page must have a skip link so the user can navigate back to the main page.
- You need to provide alternative text for each image on your webpage.
- You need to run accessibility check WAVE which is a free browser extension to check your webpages and see how they perform and provide screenshot evidence.
- Make sure use suitable colours with enough contrast, avoid green and red colour combination
-

Task 4: Website Enhancement

4.1 Features

Implement the following 3 features using appropriate HTML, CSS and JavaScript:

- Highlight the menu item in the navigator corresponding to the current webpage.

- In page order.html, create a checkbox with label "same as delivery address" somewhere appropriate near the fields for billing address. When this checkbox is checked, the fields for billing address will be filled with the delivery address automatically. If the delivery address is not completed when this checkbox is checked, display an alert or inline message: "Please enter your delivery address first".
- Allow users to choose from three types of credit card: Visa, MasterCard and American Express. Based on the type of credit card, limit the length of the credit card number, 16 digits for Visa and MasterCard, 15 digits for American Express.

4.2 Features Page

Create a fourth page named `features.html` and incorporate it into the navigator. On this webpage, list the features that you have implemented and provides hyperlinks to indicate where those features are implemented.

Submission

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

- Check your assignment against the attached marking scheme.
- Check Canvas for announcements related to the assignment;
- Ensure that all HTML and CSS files can be fully validated;
- You need to capture a screenshot of the accessibility checker for your submission
- The assignment is your original work. If your assignment is not your original work, your result will be 0 marks for this assignment;
- Provide references for all third-party contents used on your webpages;
- Test your web pages on the Mercury server. If your assignment cannot be displayed on Mercury properly, penalties will apply; and

The assignment should be submitted as an individual work through Canvas before the deadline**.**

- Upload HTML files separately then zip all CSS, JavaScript, image files and screen shot of the accessibility check into a zip file named "wd_assignment_2.zip" to submit via Canvas (In total you must have minimum of 5 files to submit).
- You need to put all the files in Mercury server and make sure everything is working then provide the link to your website as a comment in your submission.
- You can submit the files for up to 5 times through Canvas and the latest submission will overwrite the previous ones.
- Make sure you keep a copy of the assignment cover sheet.

Web Development COS10005 - Assignment 2

Marking Scheme

The assignment will be marked out of 100.

		Mark	Requirements	Deductions
Website Development	HTML Pages	15	• At least 3 HTML webpages meeting the specifications (9) • Webpages fully validated against HTML5 (3) • Source code properly indented and organised (1) • Source code properly commented (1) • At least 3 ice-cream images (1)	• Web pages not fully validated against HTML5 • Web pages not displayed properly • Not enough web pages • Lack of required HTML elements • Lack of required contents • Deprecated HTML elements/attributes used • Embedded or inline CSS • Links on web pages not working • Lack of required images • Images too large or too small
	CSS	15	• Appropriate and fluid page layout. (2) • CSS files fully validated against CSS3 (3) • CSS code properly indented and organised (2) • CSS code properly commented (1) • Consistent styles for all web pages (1) • Professional, beautiful and creative webpage style (6)	• CSS files not fully validated against CSS3 • CSS files not properly applied to web pages • Inappropriate web page layout • Inappropriate contrast between foreground and background text colour • No comments or inadequate comments • Lack of required CSS selectors • Lack of required CSS properties • Inconsistent styles for different web pages • Webpage style too simple
	JavaScript	40	• No errors in code. (10) • Fields validated as required. (18) • Form submitted only when entered data are valid. (4) • All error messages displayed in an alert or as inline messages. (4) • Code properly indented and organised. (2) • Code properly commented. (2)	• Fields not validated as required. • Form submitted with invalid input data. • Error messages not displayed as required. • No comments or inadequate comments. • Used inline or embedded script
	Accessibility	10	• Evidence of WAVE accessibility check (3) • Images having alternative text (3) • Make sure use suitable colours with enough contrast, avoid	• Missing accessibility check • Missing alternative text for images • Wrong colour combination No skip link

			green and red colour combination (2) • All web pages have a skip link (2)	
Website Enhancement	Highlighted Menu Item	5	• Feature properly implemented (5)	• Feature not properly implemented
	Automatic Input of Billing Address	10	• Feature properly implemented (10)	• Feature not properly implemented
	Adaptive Credit Card Length	5	• Feature properly implemented (5)	• Feature not properly implemented
		100		