|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Assessment # and title | | | | C-WT-AT2-POR-Phase-5 | | | | Web Technologies Portfolio Phase 5 | | | | | |
| **Lecturer name** | | | | *John Robertson* | | | | | | | | | |
| **Student name** | | | | **Hyoin Lee(Scott)** | | | | | | | | | |
| **Student ID number** | | | | **20102711** | | | | | | | | | |
| **Telephone contact number** | | | | *0432105240* | | | | | | | | | |
| **Email** | | | | **20102711@tafe.wa.edu.au** | | | | | | | | | |
| **By completing and submitting this signed form to my lecturer, I am stating that:**   1. The attached submission is completely my own work 2. I have correctly cited all sources of information used in this work (if required) 3. I have kept a copy of this assessment (where practicable) 4. I understand a copy of my assessment will be kept by the NMTAFE for their records 5. I understand my assessment may be selected for use in the NMTAFE’s validation and audit process to ensure student assessment meets requirements | | | | | | | | | | | | | |
| **Student Signature** | | *Student to fill this section out* | | | | | | | **Date** | | *Student to fill this section out* | | |
| Assessors please note: Where verbal clarification has been sought from a student to gather additional assessment evidence from an assessment item, question/s and response/s must be recorded, signed, and dated by the assessor, against the relevant assessment item/s. | | | | | | | | | | | | | |
| NB: Feedback will be given via Blackboard when possible. | | | | | | | | | | | | | |
| **Submission 1** | | Result | Satisfactory / Not Yet Satisfactory | | | | | | | Date | | |  |
| *To satisfy requirements for this assessment, you need to complete the following:* | | | Feedback to student… | | | | | | | | | | |
| **Submission 2** | | Result | Satisfactory / Not Yet Satisfactory | | | | | | | Date | | |  |
| *To satisfy requirements for this assessment, you need to complete the following:* | | | Feedback to student… | | | | | | | | | | |
| **Student Feedback** | | | Feedback from student… | | | | | | | | | | |
| Lecturer Signature | | |  | | | | Student Signature | | | | | |  |
| **Assessment type (🗹):** | | | | | | | | | | | | | |
|  | Questioning (Oral/Written) | | | |  | 3rd Party Report | | | | | |  | Practical Demonstration |
|  | Other – Project/Portfolio (*please specify on the right):* | | | |  | | | | | | | | |

Contents

[Assessment Steps 11](#_Toc149583877)

[Complete the Front Page 11](#_Toc149583878)

[Setting Up for Phase 5 12](#_Toc149583879)

[Integrate TailwindCSS into Development Workflow 16](#_Toc149583880)

[Integrate TailwindCSS into Development Workflow 16](#_Toc149583881)

[Implement Tailwind Styles 17](#_Toc149583882)

[FTP to ScreenCraft 18](#_Toc149583883)

[Testing in Multiple Browsers 20](#_Toc149583884)

[Demonstrate to Assessor 21](#_Toc149583885)

[Submission of Portfolio Work 21](#_Toc149583886)

[Appendix A: Code Style Guidelines 22](#_Toc149583887)

[Appendix B: Naming Conventions 23](#_Toc149583888)

[HTML, CSS and JS Files and folders 23](#_Toc149583889)

[File extensions 23](#_Toc149583890)

[Appendix C: Site Folder Structure 24](#_Toc149583891)

|  |
| --- |
| Assessment Due Date |
| This assessment is split into components that have several due dates.  This component of the Portfolio of Work is due:   * Week 16 at 17:30 (5.30PM) on day of scheduled lecture   These are also shown in the Learning and Assessment Plan.  Also, refer to Blackboard for most accurate dates, which may alter due to unforeseen circumstances.  We also will endeavour to update these document(s) at the same time. |
| Required Resources |
| The base requirements this assessment task, are listed below. They are listed as common (for both PC and Mac), and for the individual operating systems.  We presume that all assessment work is completed on a **PC** with the software as specified. This is to reduce configuration issues affecting the successful completion of the assessment item.  Whilst other applications and operating systems may be used, we are unable to give extensive support to ensure your environment is working as expected. Common:  * Access to Office 365 & Microsoft Word * WebStorm, PhpStorm * Access to Figma.com for diagrams * Access to TAFE Web Site Hosting (located on dev.ScreenCraft.net.au) * WinSCP for deployment of site * Git for version control, GitHub account for  PC:  * Web Browsers (Must have **at least** TWO different rendering engines)   + Chromium based: Edge, Chrome,   + Firefox   + Opera  Mac:  * Web Browsers (Must have **at least** TWO different rendering engines)   + Chromium based: Edge, Chrome,   + Firefox   + Opera   + Safari   ***Use of some of these items may not be used in this part of the assessment task.*** |
| Optional Requirements |
| An application to provide web services such as web server, database and more.   |  |  | | --- | --- | | Windows | MacOS | | Laragon  AMPPS  XAMPP  Docker | MAMP  AMPPS  XAMPP  Docker |   The lecturers will not be familiar with all the options listed, and it will be up to the student to investigate how to set up and use any alternatives.  The use of alternative IDEs is allowed (such as VS Code), but not supported. |
| Instructions |
| Follow the steps listed in this assessment item.  Submission of the documentation, code, and associated items is at the end of each part of the portfolio.  Provide evidence in the form of scanned documentation, completed assessment documents, screenshots, screencasts and other formats as required in this assessment. |
| Scenario |
| You are currently working as an intern for a small Perth-based start-up company called *Incredibly Obvious Technologies*.  The company is challenging you to work on a project to assess your skills as you develop them.  The project will build up in phases from the base content by adding semantic HTML, CSS to provide layout and suitable images to provided content.  The final section of the project is to add JavaScript to solve a specific problem.  You will find the project specifications; an outline of the phases of development and other details given below. |
| Portfolio Specifications and Development Outline |
| The following are the specifications for the project you are undertaking as part of this portfolio of work.  Each part of the project corresponds to another part of the portfolio, with each part submitted according to provided deadlines.   * Phase 0:   + Setting Up – not assessed but contains the required steps to initialise the portfolio. * Phase 1:   + Implement and test semantically and structurally correct HTML for the provided layout design and provided content. * Phase 2:   + Implement and test the required CSS to provide the layout as provided. * Phase 3:   + Add additional content and styling to amend the project to the given specifications. * Phase 4:   + Add and test a JavaScript solution for a predefined problem to the page. * Phase 5:   + Complete portfolio “dashboard” and other pages using TailwindCSS and FTP to Screencraft hosting.   Each phase will be version controlled and uploaded to the ScreenCraft Development Server.  Phase 0: Setting Up  In this phase, you are to complete the setting up of the portfolio.  This is submitted as part of Portfolio Part 1.  Phase 1: Text to Semantic HTML  In this development phase you will take the provided text and construct a suitable HTML page structure to suit it and the provided wireframes.  NO CSS is to be added.  Phase 2: Adding Structural Style  In phase 2 you now add the required CSS to style the site appropriately.  You are expected to style the site using Black, White, and Grey (#333333, #666666, #999999 and #cccccc only).  This is to demonstrate the use of CSS to lay out the site without any extra distractions such as colour and images. Use the provided layout wireframes as a guide to how it should be visually structured.  You will validate your custom CSS (not the CSS reset) using the relevant W3C validator.  Evidence of valid CSS will be submitted in the Phase 2 evidence folder.  Screenshots of the final product for this phase will also be included.  Phase 3: Pwity Pweese!  Aside: Imagine “Tweety Pie” saying Pretty Please  In this phase of the project, you are to implement the colour, typography and images to complete the basic page.  You will also amend the content by adding an extra story from phase-3.txt and an image.  In the “aside” you will add the content from phase-3-aside.txt  You will use the W3C validator to ensure your HTML is still valid.  Evidence of valid HTML will be submitted in the Phase 3 evidence folder.  Screenshots of the final product for this phase will also be included.  Once completed check in all your code to your local repository and push it to your private remote.  Phase 4: Shaking Things Up!  In the final phase, you will be implementing a simple problem that uses JavaScript.  The output of this JavaScript will be shown in the Aside area, in position 1.  The styling of the output must be in keeping with the rest of your choices of typography, colour, and other aspects of the site implementation you have undertaken.  Position 1 in the aside must contain a heading, the output from the JavaScript, and a footer. The footer of the section at position 1 must contain your name.  Phase 5: Dashboard and TailwindCSS  In this Wrap-up phase you will be showing the ability to use a CSS Framework. This framework is a Utility first system that closely resembles writing typical CSS, but using classes in a way that provides a more coder-centric system.  You will be formatting a dashboard (index.html), plus at least one other page.  You must code using TailwindCSS. |
| Site Structure Requirements |
| All files must apply the naming conventions outlined in Appendix B: Naming Conventions.  Code must apply suitable code style as advised in Appendix A: Code Style Guidelines.  The project has a defined structure outlined in the Appendix C: Site Folder Structure.  The primary page in each phase must be named index.html.  Content details and additional steps will be made more explicit in each subsequent part of the portfolio.  Ensure you DO NOT complete more than one phase of development at any time. |
| General Instructions |
| Complete each step of the document in the order given.  All work must be completed at the CLI, except for starting the development environments.  When a step requires you to confer with the lecturer, ensure you do so, and make notes as required in spaces provided in this document.  Clarifying Requirements  If you require requirements to be clarified, please follow the instructions below:  Send an email using your TAFE email address to:  [f2f@screencraft.net.au](mailto:f2f@screencraft.net.au)  You MUST include the following subject:  C4 Prog: Web Tech Portfolio Phase x  Ensure you replace the x with the phase you are asking about. In this case 3.  Online Students must use the [online@screencraft.net.au](mailto:online@screencraft.net.au) email address for these forms of questions. |
| Answering Questions |
| When a step includes a question, you must attempt to answer it.  There is a minimum and maximum number of words to use for each answer.  Any step that requires answers to be provided will have a space in this document immediately after each step. The answer space will expand with the content you type or images you paste.  Resize images to fit the space provided, ensuring they are still legible.  Do not include a whole screenshot, just the required details.  If a step has more than one question, these maxima and minima are a total for all the questions in that specific step.  All answers must be in complete sentences unless indicated.  *If required, make sure to add any code you’ve written in a separate file to your submission.*  *DO NOT put long pieces of code (over 10 lines) in this document.* |
| Sources of Information |
| In industry, it is good practice to keep track of where information was obtained. This is especially true if it is a written document, or even code.  If you answer any questions using information from web sites, please include the site name and URL (Web site address) after the answer.  Likewise, include the title and author for books and magazine articles.  For example:   * RS Electronics Ltd:  <https://au.rs-online.com/> * Slack API Documentation, Users List Method:  <https://api.slack.com/methods/users.list> |
| Code Storage |
| We advise that you create a private GIT repository on GitHub and use this to store a copy of your work.  This assessment has instructions on setting up version control and a remote (private) repository. Do not initialise any version control until told to do so.  To ensure you have a recent copy of your code you **must** use **AT LEAST TWO** of the following methods, of which a version-controlled copy is REQUIRED:   * Private Git repository,   and one of:   * Cloud Storage (OneDrive within your college Office365), and/or * Keep a copy on a USB thumb drive.   Backing up to One Drive or to USB is best done by compressing the project folder up before copying to either OneDrive or USB. |
| Code Style |
| Please see **Appendix A: Code Style Guidelines** for details on the code style(s) to use for the assessment. |
| Professional Code and Documentation Requirements |
| This document contains appendices that cover the professional requirements when creating files, code, project structures and more.  These requirements must be adhered to.  You are only allowed to use TailwindCSS and FontAwesome Free during this phase. Configuration and other details are given in this assessment.  This means that you may not employ frameworks such as Bootstrap, jQuery, et al in your work. |
| Assessment Structure |
| To complete this assessment, you will be asked to complete five parts.  Do not start subsequent parts until completing and submitting previous part.  These parts include:   * Part 1: Development Phases 0 & 1, Setting up and creating semantic markup. * Part 2: Development Phase 2, Adding basic styling to create desired page layout. * Part 3: Development Phase 3, Adding further style, content, and imagery to enhance page. * Part 4: Development Phase 4, Adding JavaScript to solve a given problem and display result on page. * Part 5: Development Phase 5, Dashboard using TailwindCSS. |
| Assessment/Development Process |
| In the steps that are contained in this document, you will design several pages, and create the layout for the home page, with your code being version controlled as you progress.   * Set up. * Integrate TailwindCSS into development workflow. * Implement Tailwind Styles to produce a dashboard for the portfolio of work. * Test HTML is displayed as expected using at least two different browsers with different web rendering engines. * Compress and submit your work. |

# Assessment Steps

Complete the steps in the order given.

| **STEP** | **Task to perform** |
| --- | --- |
| 00 | Complete the Front Page Make sure you have filled out the front page of this document.  Familiarise yourself with the content and document your progress in this assessment. This means, **READ** the **WHOLE** assessment **AT LEAST ONCE** before starting any work.  Make notes on the requirements of this assessment, as details appear as you progress through and are not given in one location.  This is very similar to how details emerge during the continuous development interviews and meetings with a client.  At any stage during this assignment, you may consult the stakeholder(s) or their representative(s).  Max Sentences: N/A |
|  | Intentionally left blank |
| 01 | Setting Up for Phase 5 Open the current project in your web editor, be that WebStorm, VSCode or such.  We will need to use a terminal for this stage of our development process. For this, WebStorm has a built-in terminal, VS Code has a Terminal extension that may be installed, or you may use the Terminal from Laragon when on a PC.  We are presuming that you have stored your portfolio in the location c:\Users\USERNAME\Source\Repos\C4Prog-WT-POR\ on a Windows machine or in the /home/USERNAME on a Linux machine or /Users/USERNAME folder on a MacOS system.  You will need to modify some commands to suit if you have not placed the portfolio in the appropriately listed location.  We are also presuming the capital letters are as shown, so any differences will need to be made at the time of entering the commands for this step. WebStorm: Open the terminal tab by locating it at the bottom of the screen and clicking on the Terminal tab.  If you have a Windows machine it is advisable to open a new CMD prompt terminal rather than a PowerShell (PS) one. To do so use the V down arrow next to the + sign on the terminal area, and then select CMD.  If you have a newer version of WebStorm the terminal may not be present, so use the keyboard shortcut of: ALT+F12 / Option+F12. VSCode: Install the Terminal extension.  Open the terminal using… Getting into the Correct Folder As stated above, we presume that you have your project in the Source\Repos\C4Prog-WT-POR folder.  To get into the correct folder use:  Windows: cd %userprofile%\Source\Profile\C4Prog-WT-POR  MacOS/Linux: cd ~\Source\Repos\C4Prog-WT-POR  Once the terminal is open, and you are in the correct folder, verify that the following are available on your system by using the following commands:  NodeJS: node --version  NPM: npm --version  NPX: npx –version  Any version of NodeJS after 16 is suitable, and any version of NPM/NPX after 8 should also be suitable.  If an error occurs it means that Node and NPM are not installed. You will need to rectify this before continuing. Node is Ready… Given that Node is available, and NPM/NPX are also working, we now start the TailwindCSS installation and “development” process.  Enter the following commands:  Install Tailwind: npm install -D tailwindcss  Initialise Tailwind: npx tailwindcss init  In your editor search for and edit the tailwind.config.js file which will be in the root of the project.  Make sure that the lines in this file now read:  */\*\** ***@type*** *{import('tailwindcss').Config} \*/* module.***exports*** = {  content: [  './index.html',  './phase-\*/\*.html',  './pages/\*\*/\*.html',  ],  theme: {  extend: {},  },  plugins: [], };  This configuration scans the index.html file in the root of the project, the various HTML files in phases 1 to 5, and the pages folders for any HTML files that contain any Tailwind Classes in the code. Create an Input CSS file Next create a folder called src in the root by using the command:  mkdir src  Then, in your editor create a new CSS file in this src folder and call it “input.css”. Case will be important in following commands.  Now execute the following command (this is a SINGLE LINE):  npx tailwindcss  -i ./src/input.css  -o ./assets/css/site.css  --watch  The command is on a single line, but to make it easier to read it is shown split over multiple lines.  Open the index.html file from the root folder of the project and add in the HEAD section of the document:  <link rel="stylesheet" href="./assets/css/site.css">  Go to a line with  <h2>Certificate IV in IT (Programming)</h2>  Edit this line by adding the following classes to the h2 element:  <h2 class="text-xl font-bold text-gray-500">  Open the page with your browser using the editor’s built in web server.  You should see something like this:    You are now ready to continue with the portfolio.  Use a commit message based on the one below (note this is incomplete):   * “feat(phase-5): TailwindCSS setup”   Push the code to your private remote repository.  Max Words: N/A |
| 02 | Integrate TailwindCSS into Development Workflow In this phase we want you to investigate how the sample code provided at <https://components.dev.screencraft.net.au> is structured. Pay attention to some of the methods used on the copyright page as this shows a header, footer, main area, and simple cards used for the references.  **Important:** This site is in ongoing development and will change on a frequent basis.  Not all components will be immediately available.  Please log a request for a specific component ScreenCraft Helpdesk with the message subject “SCC Component Request”.  This site contains some examples of how common components may be formed. For example, these may include:   * A sticky header, * A “vertical card” with a header and footer and content, * A “horizontal card” with an image (on left) and content (on right), * A navigation bar, * A footer, * Centre content on screen, * And many more as the site expands.   You will need at least some of these to complete the dashboard.  Identify the “components” you will use on your dashboard.  Add the details of these components to your site ReadMe.md and in the answer area provided.  Use a commit message based on the one below (note this is incomplete):   * “feat(phase-5): Component identification”   Push the code to your private remote repository. |
| A02 | Integrate TailwindCSS into Development WorkflowWhat components will you need for your dashboard? Sticky header, vertical card, horizontal card, navigation bar, footer |
| 03 | Implement Tailwind Styles In this phase you will be adding the TailwindCSS classes to the dashboard code.  We strongly suggest you complete this in the following order, and after each is complete, commit your code. To assist we also suggest possible commit messages for you.   |  |  | | --- | --- | | Header | feat(phase-5): Implement header using TailwindCSS | | Navigation | feat(phase-5): Implement navigation  within header  or  feat(phase-5): Implement navigation  after header | | Footer | FEAT(phase-5): Implement footer | | One “card” for dashboard main area | feat(phase-5): Implement test card  for dashboard | | Main area - including remaining cards | feat(phase-5): Implement dashboard | | Copyright page | feat(phase-5): Add copyright page  You may utilise the provided page as a guide for your copyright page. | | Tidy Up |  |   After all steps above are complete, and the dashboard is ready, perform a final commit and push the HTML to your private remote repository with a suitable commit message. |
| 05 | FTP to ScreenCraft Select a way to connect to ScreenCraft for your Portfolio of work.  You may use any of the following applications:   * WinSCP (Windows) * WebStorm (Cross platform) * PhpStorm (Cross platform) * Cyberduck (MacOS)   Configure the FTP client, taking screenshots as the client is configured.  You will have obtained these details via email from the ScreenCraft Admin (Adrian Gould).  Add the screenshots to the evidence folder and to the space provided in A05.  Once the configuration is completed, test by uploading just the index.html (dashboard) file from your site.  Head to your website on the ScreenCraft Development Server and take a screenshot of what is shown.  Your development server website address will be of the form:  <https://dev.screencraft.net.au/~WEB_USERNAME>  Copy the screenshots into the evidence folder inside phase-5.  IMPORTANT: Whilst performing the next part, you must take screenshot(s) of the site upload in action.  Now upload the remaining files but NOT the following files and/or folders:   * node\_modules * .git * …   Refresh your site in the browser and take a screenshot of what is shown.  Copy the screenshot into the evidence folder inside phase-5.  Ensure the evidence folders are part of your “.gitignore” file in the repository.  Commit and push the update to your private remote repository with a suitable commit message:   * Feat (phase-5): FTP configuration and upload |
| A05 | Add screenshots of your FTP Configuration  Add screenshot(s) of site upload being performed   Testing by uploading just the index.html file to dashboard.  Src, phase-5, phase-4 uploaded  Phase-3 uploading    Phase-2, phase-1, assets uploaded    All files uploaded |
|  |  |
| 06 | Testing in Multiple Browsers The final step in this portfolio phase is to test your dashboard and portfolio in at least TWO different browsers.  You must select **at least one browser** from **two different rows**.  We have not listed Linux browsers, so if you are testing on a linux system, please verify your selection with your lecturer.   |  |  |  | | --- | --- | --- | | Operating System ➡ Engine ⬇ | Windows | MacOS | | Blink | Chrome  MS Edge  Brave  Opera | Chrome  MS Edge  Brave  Opera | | Webkit | Otter Browser | Safari | | Gecko | Firefox | Firefox |   Which Browsers did you select? Add the details to the ReadMe.md and to the space provided in A06.  Commit and push the update to the updated file to your private remote repository with a suitable commit message:   * Feat (phase-5): Final testing |
| A06 | Which Browsers did you select? Google Chrome  MS Edge |
| 07 | Demonstrate to Assessor Demonstrate to the assessor.  Obtain their feedback, update your work as required based on their feedback, ensuring all changes are notes as semantic commits using the “fix:…” tag at the start.  Prepare for phase 5 of the site development. |
| END | Submission of Portfolio Work To submit the portfolio, do the following:   * Save this document with your answers. * Export the pages of your Figma file to PDF(s) * Compress the portfolio folder using 7-Zip * Open Blackboard, locate and open the “AT2-PT1 Por Phase 4” item * Upload this completed assessment document. * Upload the PDF and Word document(s). * Upload the compressed site files. * Click submit.  All answer documents MUST be submitted in Microsoft Office 365’s Word format. It is important that the PDF and Word documents are separate from the portfolio compressed folder as this makes it much easier for assessment. |

# Appendix A: Code Style Guidelines

The following guidelines should be applied to your code as it is developed.

Many may be applied via the use of PhpStorm, PyCharm or similar plugins and code formatting.

### PHP Code (General)

Please refer to the PHP PSRs:

* <https://www.php-fig.org/psr/psr-1/>
* <https://www.php-fig.org/psr/psr-12/>
* <https://www.php-fig.org/psr/psr-4/>

### Applications Built with Laravel

Please refer to the following articles:

* <https://dev.to/lathindu1/laravel-best-practice-coding-standards-part-01-304l>
* <https://dev.to/lathindu1/laravel-best-practice-coding-standards-part-02-a40>

### HTML Code

Please refer to the Google Style guide:

* <https://google.github.io/styleguide/htmlcssguide.html>

Note that PhpStorm and WebStorm will format code to their own standard, which is acceptable.

### JavaScript

Refer to the Google JS Style guide:

* <https://google.github.io/styleguide/jsguide.html>

Note that PhpStorm and WebStorm will format code to their own standard, which is acceptable.

### Python Code

Your code will follow the PEP 8 standard.

PyCharm will format code very close to the PEP-8 standard.

### JSON Code

JSON should be formatted in an appropriate manner.

*Readability Counts   
- Zen of Python*

# Appendix B: Naming Conventions

## HTML, CSS and JS Files and folders

* No spaces.
* Alpha-numeric characters only except minus/hyphen (-) and full stop (.).
* All lowercase letters.
* Use hyphens (dash) between words.

## File extensions

The following file extensions must be used:

|  |  |
| --- | --- |
| HTML pages | .html |
| JS | .js |
| CSS | .css |
| PHP | .php |
| Node | .js |
| React | .jsx |
| Python | .py |
| JSON | .json |
| XML | .xml |

# Appendix C: Site Folder Structure

The diagram below outlines a folder structure for the portfolio from Phase 1 to Phase 4:

# Appendix D: Useful Resources

The following links may be of use when developing this stage of the portfolio.

* <https://components.screencraft.net.au>
* <https://tw-elements.com/>
* <https://www.hyperui.dev/>
* <https://merakiui.com/>
* <https://wickedblocks.dev/>

Also check <https://diigo.com/user/Ady_Gould> and search for TailwindCSS.

We STRONGLY recommend that you DO NOT mix and match component libraries.

We STRONGLY recommend that you DO list the resources used in the Copyright Page.

We STRONGLY recommend that you modify the source code provided by any resource to be semantically well structured. For example, replace a div that is being used for a navigation with the appropriate semantic element.