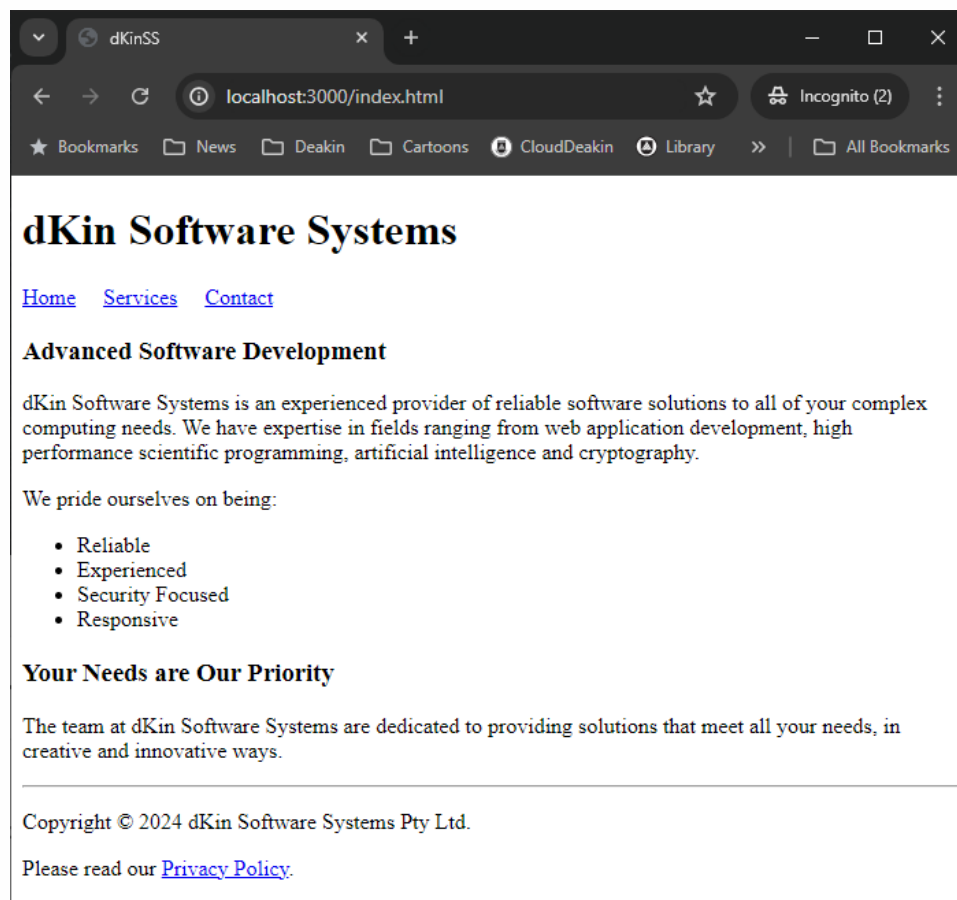


2.3P: Create, validate and publish a structural web page

Tasks

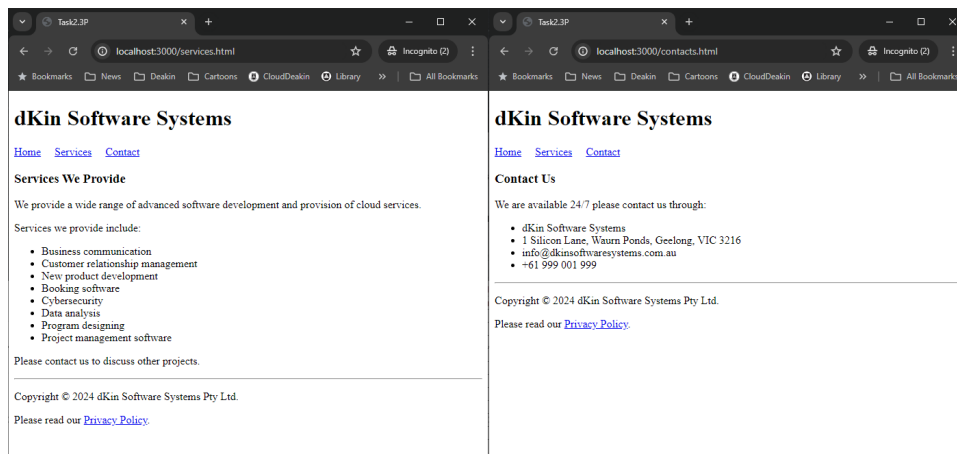
Step 1

1. Create a **dKin Software Systems** home web page (`index.html`) like the one below, using HTML5 structural elements.



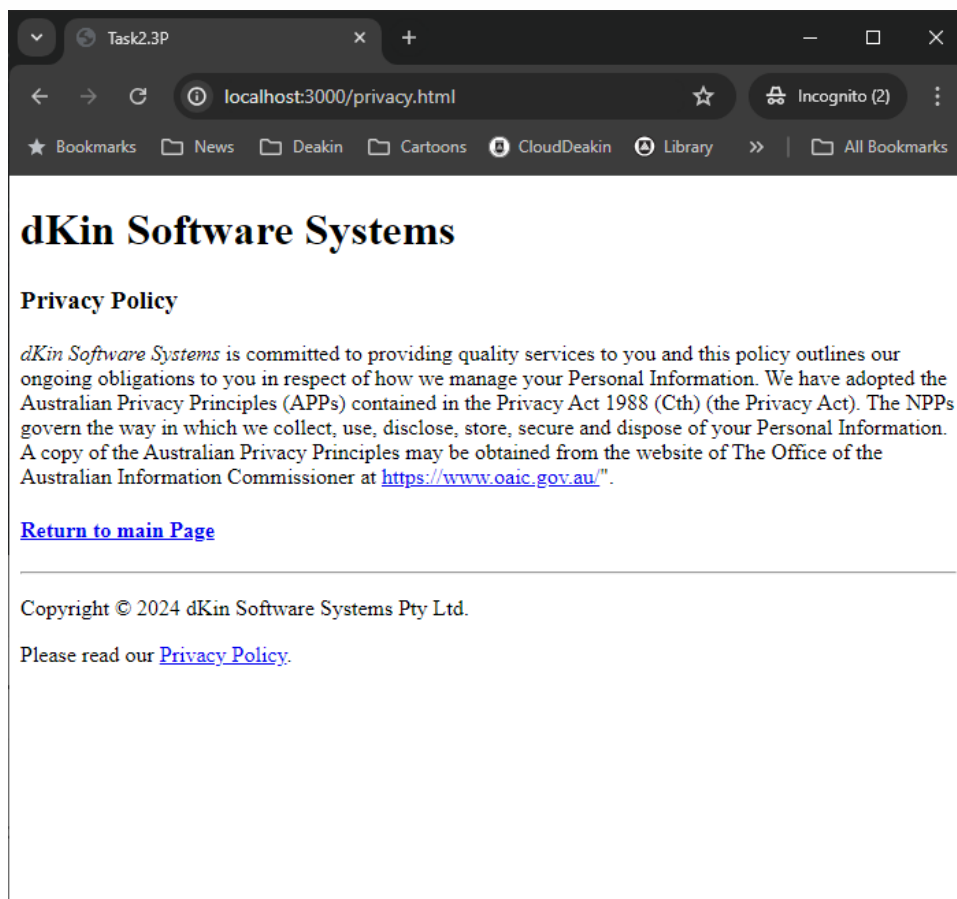
Sample dKin Software Systems Home Page

2. For the hyperlinks "*Services*" and "*Contact*" in the **dKin Software Systems** home page, create two **new** corresponding web pages "`services.html`" and "`contact.html`", and save them into the **same folder** as the "`index.html`" file. Each web page should keep the same structure of the home page, e.g., each page should have a "**header**" section, three hyperlinks "**Home**", "**Services**" and "**Contact**" should always be in the **nav** section of the page, and the content in the **footer** section should be consistent across the web pages.



dKin Services & Contact Pages

The link in the footer should also take you to a fourth page "privacy.html" file which should display some information as shown below:



dKin Privacy Page

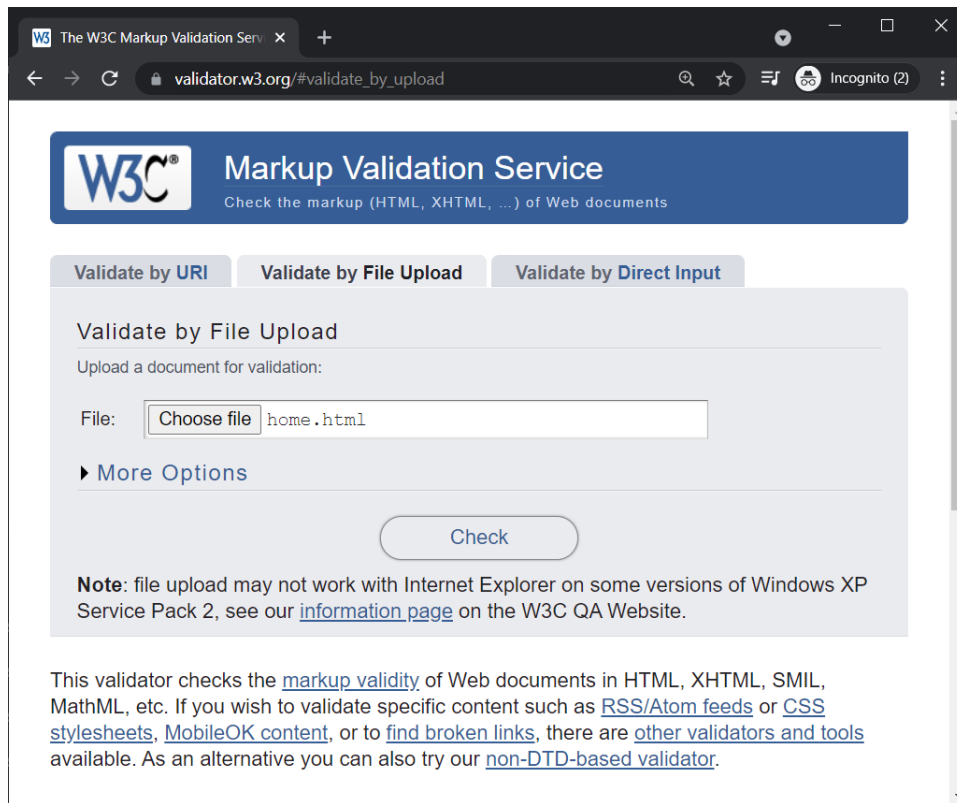
3. Check hyperlinks in each web page to make sure they link to the correct web pages when clicking them.

Step 2

The W3C free Markup Validation Service, which is available at <http://validator.w3.org>, provides you with quick self-assessment by validating your HTML code and checking it for syntax errors. The validation serves as a

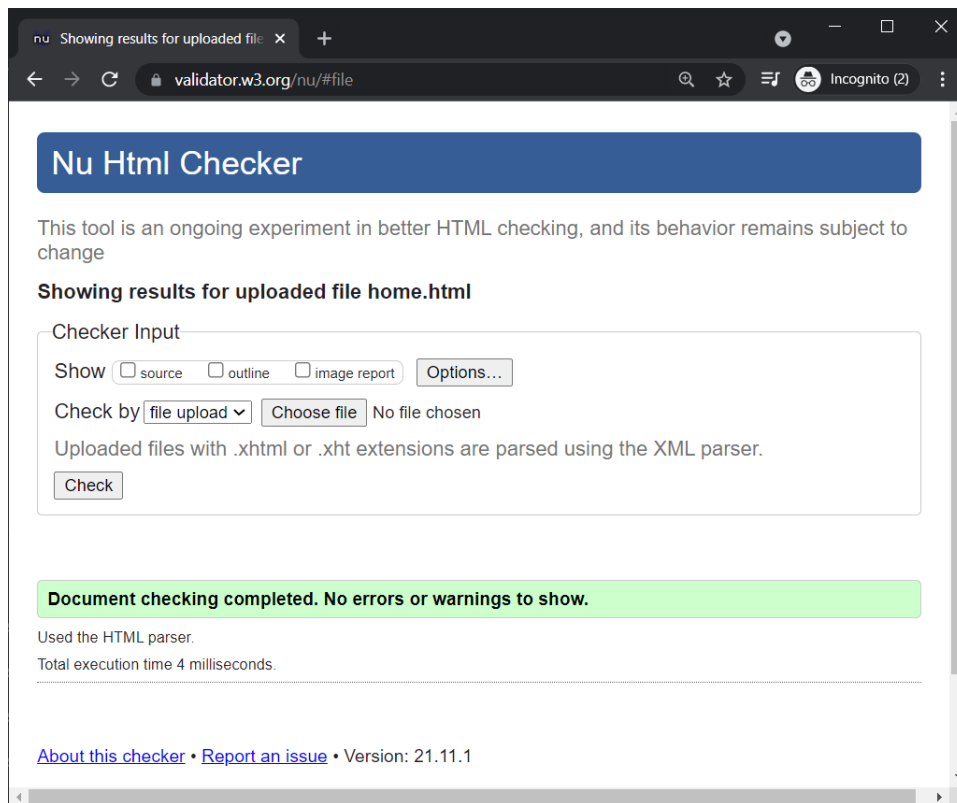
quality assurance tool, as invalid code may cause browsers to render the pages slower than otherwise.

1. Launch a browser (e.g., Firefox, IE, or Chrome) and visit the W3C Markup Validation Service at <http://validator.w3.org>.
2. Select the tab "validate by File upload" as below:



Validation

3. Click the "Browse..." button, and select an HTML file/web page you created before in previous tasks for validation. Then Click the "check" button to upload the file to the W3C validation site.
4. Check the validation results returned by the validator, and open your HTML file in a text editor to fix the problems identified by the validation.
5. Validate the problem-fixed HTML file again until no problems reported by the validator.



Validation

Step 3

1. **Publishing to the World** - making your web pages available to the world via *Deakin webserver*.

Where is the Deakin webserver?

- The Deakin webserver is mapped as the "**public_html**" folder in **your home directory** (ie H: drive) on Deakin's home network server.
- Your home directory is automatically set up each time you log into a Deakin workstation, including those in teaching spaces. So you DO NOT need to map your home directory when using a Deakin workstation on campus (e.g., in the computer labs on campus at Deakin).
- You can also access your home directory from your own personal laptop or computer at home. To do this you must first [establish a secure \(VPN\) connection to Deakin using CISCO AnyConnect Software](#), then follow the [instructions to map your home directory \(Windows\)](#), for Mac users follow the [instructions \(macOS\)](#).

How to publish your web pages on the Deakin webserver?

- Locate the "**public_html**" folder on the H: drive. Create a subfolder, for example "**sit774**", in the "**public_html**" folder.
- Place your web pages (HTML files), as well as other related files such as images, in the subfolder (e.g., *sit774*).
- Your web page (eg *home.html*) then can be accessed via a web browser

from the URL address like:

■ `http://www.deakin.edu.au/~username/sit774/home.html`

Please note, the "*username*" in the URL must be replaced by your actual Deakin username. For example, if your Deakin username is *johnsmith* and you have stored the "*home.html*" file in the subfolder as above, then the URL would be:
`http://www.deakin.edu.au/~s240123456/sit774/home.html`

Note

- You do not need to include the "`public_html`" as a directory in the URL, as it is what the tilde (`~`) is for. The tilde symbol (`~`) 'points' the server to your `public_html` area.

2. **Publishing Locally** - making your web pages available via your *local NodeJS webserver*.

- Your local NodeJS web server is located in the directory to which you extracted the "*Node.js starter template*" (more details are in the article 1.7 of week 1). Here we suppose your NodeJS web server is in "`C:\\sit774_starter_template`".
- Locate the "`public_html`" folder on the server (there should be the `index.html` file in it). Create a subfolder, for example "`sit774`", in the "`public_html`" folder (i.e., `public_html\\sit774`).
- Place your web pages (HTML files), as well as other related files such as images, in the subfolder (e.g., `sit774`).
- Your web page (e.g., *index.html*) then can be accessed via a web browser from the URL address like:

■ `http://localhost:3000/sit774/index.html`

Or, simply:

■ `http://localhost:3000/sit774/`

What will you submit?

You should submit:

- The `html` source files of *four* web pages `index.html`, `services.html`, `contact.html` and `privacy.html`.
- Screenshots of the four page validations, along with the returned results from the validator.
- Screenshots of your web page accessed via the URL addresses (via Deakin web server **and** local NodeJS server) in a browser.