MONASH University

FIT5032 - Internet Applications Development

Background Topics & History

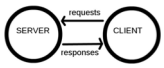
Prepared by - Jian Liew

Updated by ABM Russel

How the Web Works

This slide aims to provide a **simplistic view** of what happens when you view a webpage in a web browser on your computer or phone.

Computers connected to the web are called clients and servers.



The clients are usually the user's internet connected devices and web accessing software on those devices. This is usually a **web browser** like "Internet Explorer", "Chrome", "Firefox", "Edge" or the lesser known Opera.

The servers are computers that **store web pages, sites or apps**. When a client device wants to access a webpage, a copy of the webpage is downloaded from the server onto the client machine to be displayed on the users web browser.

In addition to the client & server...

These are important as well.

- **Internet Connection** - Allows you to send and receive data on the web.
- **TCP/IP** - Transmission Control Protocol and Internet Protocol are communication protocols that define how data travel across the web.
- **DNS** - Domain Name Servers are address book for websites. (For example, a famous DNS server would be 8.8.8.8 and these days it is 1.1.1.1)
- **HTTP** - Hypertext Transfer Protocol is an application protocol that defines a language for clients and servers to speak to each other.
- Component Files - Two main types.
  - Code Files: Websites are primarily built from HTML, CSS, and JavaScript. For FIT5032, you will learn how to use C# as well. (Together, with it you will be introduced to the .NET framework)
  - Assets: This is the collective name for the other stuff that makes up a website, such as images, music, video, word documents, pdf and etc.

Our server side code will be done using .NET (C#)

So what happens?

When you type a web address into your browser, for example www.monash.edu:

1. The browser goes to the DNS server, and finds the real address of the server that the website lives on.
2. The browser sends a HTTP request to message to the server, asking it to send a copy of the website to the client. This message is sent across your internet connection using TCP/IP.
3. The server approves of the clients request, and sends a 200 OK meaning that the client can take a look at the website.The server then starts sending packets to the client.
4. The browser assembles these packets and complete them into a website and displays it to you.

HTML Standards

HTML is constantly undergoing changes and development.

Just as there are standards for programming languages, HTML has standards too.

- HTML standards is defined by a group of people drawn from various groups interested in the development of the web. This is called working group.
- One of the main players is the **World Wide Web Consortium (W3C)**, however in the recent years, the **Web Hypertext Application Technology Working Group (WHATWG)** has taken over in popularity as with any ***competing standards*** there are often **disagreements within interested parties**. The WHATWG has popularized their ***Living Standard*** which has become one of the go to standards in the recent years.
- The main reason this happened is because of a ***disagreement*** regarding how HTML5 standards should evolve.
- In recent news, Apple, Mozilla and Microsoft VS W3C.
- Also, please note that the W3C working group has no relation with w3school.

HTML Development Levels

HTML Level 1.0
HTML Level 2.0
HTML Level 3.0
HTML Level 4.0
HTML Level 4.1
HTML Level 5 ⇦ We will be using this
Mozilla (HTML With Netscape extensions)
XHTML Level 1.0
XHTML Level 1.1
XHTML Level 2.0
HTML Living Standard ⇦ By the WHATWG group

Relevant XKCD

XKCD is a webcomics which subject matter varies from statements of life and love to mathematical, programming and scientific jokes.

What people think……

When there are two versions of a spec, what's a programmer to do? There's no one answer. ***"Generally we tell developers to look at the WHATWG version, which tends to be developed with better technical accuracy,"*** advised Mozilla Chief Technology Officer Andreas Gal.

"If you want to see what's already implemented in browsers now, look at W3C spec," said Opera's Lawson. **"If you want to see what might be coming (or how things may change) look at WHATWG spec."**

What the FIT5032 Teaching Team thinks…

We will follow the W3C spec, however it is good to at least know what is happening in the industry and what the WHATWG specification is.

W3School

Good or bad?

Most people that have attempted to learn web programming would have heard of w3school but...

One of the more famous resource to learn web development is www.w3schools.com

However, often times, experienced and seasoned developers will not use it as the go to reference point.

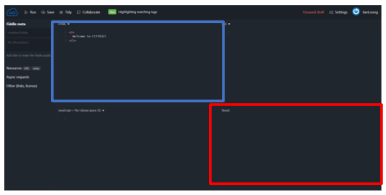
The reason is because, **previously** w3schools has been known to have poor documentation and have many content errors and issues. (This was back in 2011). They have improved since then however there might be still some issues (Some developers will find it quite obvious where the mistakes are).

In our opinion, for beginners, w3schools has fairly good tutorials that offer a good learning experience, however a more complete education will involve using more reputable resources like **Mozilla Developer Network (MDN)**. Link here.

MDN will be one of your go to reference point for FIT5032 - Internet Applications Development. If possible, please avoid using w3schools.

Online Source Code Playground

- In order to demonstrate the demos in this slides, JSFiddle will be used.
- JSFiddle is an online source code playground that can be used to test HTML, CSS or JavaScript.
- You can also copy and paste the example code to the HTML input on the site and hit run to see the results.



Structuring the Web

HTML – **H**ypertext **M**arkup **L**anguage

What is HTML?

- HTML is code that is used to structure a web page and its content.
- Contents could be structured within a set of paragraphs, a list of points or using images.
- HTML is **NOT** a programming language. It is a **markup** language that defines the structure of your content. There is also a markdown language.
- HTML consist of a series of **elements.** These elements are enclose or wrap different parts of contents to make it appear or act a certain way.
- For example, the following line…

```
1 | Welcome to FIT5032.
```

We could specify that it is a paragraph by enclosing it in paragraph tags.

```
1 | <p>Welcome to FIT5032.</p>
```

HTML elements must always be enclosed within tags.

Structuring the Web

HTML – **H**ypertext **M**arkup **L**anguage

Why is HTML not a programming language?

- Programming languages have functional purposes. These aim to achieve a programming logic.
- HTML does **not** contain programming logic.
- It does **not** contain conditional statements such as if and else.
- It **cannot** evaluate any expression or do any calculations.
- It does **not** handle events or carry out tasks.
- You **cannot** declare variables or write functions.
- It does **not** modify or manipulate data.
- It **cannot** take input or produce output.
- HTML is used to **mark up text** for the browser to read and interpret.
- It is a **markup** language.

Anatomy of an HTML element

HTML – Hypertext Markup Language

Please note that HTML5 tags are not case sensitive but we will always use lower cases. This is because of the **Code Guide** we follow. Our Code Guide will be shown later.

What is HTML <p> element?

```
<p>Welcome to FIT5032.</p>
```

The main parts of our element

1. The **opening tag**: This consist of the name of the element (in this example, it is a p) wrapped in opening angle brackets. This states where the element begins, or starts of take effect. P stands for paragraph.
2. The **closing tag**: This is the same as the opening tag, except that it includes a forward slash before the element name. This states where the element ends. Failing to include a closing tag is one of the common beginner errors and can lead to strange results.
3. The **content**: This is the content of the element. In this case, the content is Welcome to FIT5032
4. The **element**: The opening tag, the closing tag and the content together comprise the element.
5. You can preview it here.

Anatomy of an HTML element

HTML – Hypertext Markup Language

Attributes?

```
<p class="example">Welcome to FIT5032.</p>
```

Elements can also have one or more attribute.

Attributes contain extra information about the element you do not want to appear in the actual content. Here, class is the attribute **name** and example is the attribute **value**.

An attribute should always have

1. A **space** between it and the element name (or the previous attribute, if the element has one or more attributes)
2. The attribute name followed by an **equals** sign.
3. Opening and closing **quote marks** wrapped around the attribute value.

Our code guide tells us to use double quotes instead of single.

MONASH University

Anatomy of an HTML element

HTML – Hypertext Markup Language

An example of an element with two attributes.

The anchor element

```
<a href="https://www.google.com" title="Google">Click me to go Google.</a>
```

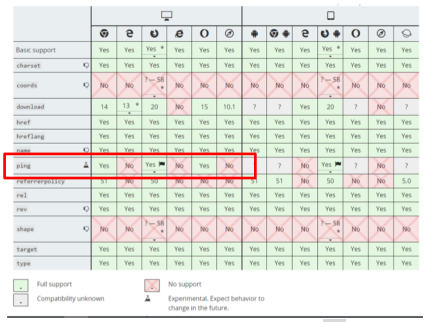
This element creates a hyperlink to other web pages, files, locations within the same page, email address or any other URL.

- href → Hypertext Reference. This contains a URL or URL fragment that the hyperlink points to. So, the value of this attribute is the web address you want go to when the link is clicked. Thus, upon clicked, you will navigate to https://www.google.com It is important to specify the protocol (http or https) so that you always get the expected results.
- title → A **global** attribute, contains text representing advisory information, related to the element it belongs to.
- For more information regarding this element. Please refer here.
- JSFiddle link can be found here.

You can also link to a non HTML resource like a PDF.

A link title is only available on mouse over.

In HTML5, title can be used at any element unlike previously.



Browser compatibility for the attributes of the anchor `<a>` element

Browser Support

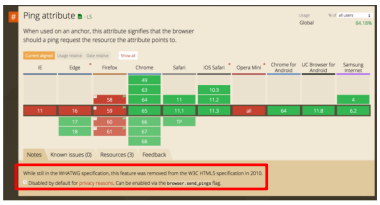
It is important to understand **different browsers** have **different support** for attributes.

For example, from the table, Internet Explorer and Edge does **not** support the ping attribute in the anchor element. Firefox on the other hand offers "questionable" support.

So, it is important for **cross-browser testing** to be done.

This table is obtained from MDN.

MONASH University



Can I use it?

This information can also be obtained by using the web site www.caniuse.com.

"Can I use" is a site that provides up to date browser support tables for support of front-end web technologies on desktop and mobile browsers.

In this scenario, you would realise that the ping attribute has been removed from the W3C specification whilst it still exist in the WHATWG specification.

This is due to competing standards.

MONASH University

Anatomy of an HTML element

Single Quote or Double Quote?

For this unit, all HTML attributes are wrapped using **double quotes**. However, if you are using materials from online sources, you will sometimes see single quotes.

This is a matter of personal preference. But, for this subject, we **prefer double quotes** over single quotes.

```
1 | <a href="https://www.monash.edu/">Monash University Link</a>
```

Preferred Way for FIT5032

```
1 | <a href='https://www.monash.edu/'>Monash University Link</a>
```

Avoid this

However, you should **never** mix double quote and single quotes.

```
1 | <a href='https://www.monash.edu/">Monash University Link</a>
```

Do not do this.

But if you already used one type you can use the other type

```
1 | <a href="https://www.monash.edu/" title="Isn't Monash Great?">Monash</a>
```

OK

You can also omit the quotes. But please do not do that. :(

MONASH University

Anatomy of an HTML element

HTML – Hypertext Markup Language

Nesting elements

You can put elements *inside* other elements. This is called *nesting*. For example, we could wrap the word FIT5032 in a `<strong>` element, which means that the word is to be strongly emphasized.

```
<p class="example">Welcome to <strong>FIT5032</strong>.</p>
```

Correct way.

It is important to ensure your elements are properly nested. In the example given, the `<p>` element comes first followed by the strong, there we must close the `<strong>` before the close the `<p>` element. (The order is important)

```
<p class="example">Welcome to <strong>FIT5032</p>.</strong>
```

Avoid doing this.

It is important to open and close the elements correctly. This is to ensure the web browser will behave as expected.

MONASH University

Anatomy of an HTML element

HTML – Hypertext Markup Language

In HTML5 it is not needed to put a closing tag for empty elements. (Self Closing)

Empty elements

Some elements have no contents. These are called *empty elements* or *void elements*.

CORRECT WAY

```
<img src="http://via.placeholder.com/350x150" alt="A placeholder image">
```

In this example, it contains ***two*** attributes but it does not contain a `</img>` and it does not have inner content. This is because an image element does not wrap content to have an effect on it.

HTML5 does **not** require empty elements to be closed.

The purpose is to embed an image in the HTML page in the place where it appears.

AVOID DOING THIS!

```
<img src="http://via.placeholder.com/350x150" alt="A placeholder image"></img>
```

MONASH University

Anatomy of an HTML element

HTML – Hypertext Markup Language

Image tag in more detail..

One of the more important aspects of a website, is also its **accessibility**. This will also be one of the topics covered in this unit. Let's take a deeper look...

```
<img src="https://en.wikipedia.org/wiki/File:Monash-shield.png" alt="Monash Logo">
```

Remember the alt attribute

The purpose of the src attribute here is to

- specify the path to our image file

The purpose of the alt attribute here is to

- specify ***descriptive*** text for users who cannot see the image which might be due to ***visual impairment***
- If something went that causes the image not to display, the descriptive text will be shown instead.
- It is always a good idea for the alt value to provide enough information so that the reader have a good idea what the image conveys

MONASH University

Boolean Attributes

It is possible to have attributes without values. These are known as boolean attributes.

They can only have one value, which is the normally **the same** as the attribute name.

```
<input type="text" disabled="disabled">
```

This is the same as

```
<input type="text" disabled>
```

This is called the **shorthand notation** or the **minimised form**. In short, the appearance of this value denotes that it is set to true and their **absence implies that the value is false.**

In HTML5 it is not necessary to close input tags. They are empty elements. It must have a start tag and must not have an end tag.

MONASH University

The list element

There are two basic types, **ordered** and **unordered** list

Unordered

Unordered lists are used to mark up lists of items for which the order of the items doesn't matter

```
<ul>
  <li>milk</li>
  <li>eggs</li>
  <li>bread</li>
  <li>hummus</li>
</ul>
```

There is `<li>` inside of the `<ul>`

Ordered

Ordered list are list in which the order of the items does matter. A good example of this would be a set of directions. (This will generate numbering)

```
<ol>
  <li>Drive to the end of the road</li>
  <li>Turn right</li>
  <li>Go straight across.</li>
  <li>Monash University is on your right.</li>
</ol>
```

It is also possible to do a nested list. (Not shown here)

MONASH University

The Description List element

First defined in HTML4.01

The `<dl>` element represents a description list

The element encloses a list of group of terms using the `<dt>` element and descriptions `<dd>`. A simple use case for this would be to display a glossary section.

```
<dl>
  <dt>Faculty of IT</dt>
  <dd>
    Most awesome faculty in Monash.
  </dd>
  <dt>Faculty of Business</dt>
  <dd>
    Business Faculty.
  </dd>
</dl>
```

MONASH University

Anatomy of an HTML document

The Doctype for HTML5

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>FIT5032 Internet Applications Development</title>
  </head>
  <body>
    <img src="https://en.wikipedia.org/wiki/File:Monash-shield.png" alt="Monash Logo" >
  </body>
</html>
```

`<!DOCTYPE html>`

The doctype. This defines a set of rules that the HTML page had to follow to be considered good or valid HTML. Each version would use a different doctype.

MONASH University

Anatomy of a HTML document

<html></html>

The <html> element. This wraps all the content on the entire page. This is known as the root element.

<head></head>

This element acts as a container for all the stuff you want to include on the HTML page that is not the content you are showing to your viewers.

<meta charset="utf-8">

This element sets the character set your document should use to UTF-8, which includes most characters from the vast majority of human written languages. (This is important for non-roman characters like Chinese or Japanese)

<title></title>

The <title> element. This sets the title of your page, which is the title that appears in the browser tab the page is loaded in, and is used to describe the page when you bookmark/favorite it.

<body></body>

This contains all the content that you want to show to web users when they visit your page, whether that's text, images, videos, games, playable audio tracks, or whatever else

MONASH University

Marking up text

Headings

Headings elements will allow you to specify that certain parts of your contents are headings or subheadings of your content. Imagine a book that has a main title, chapter title and subtitles. HTML contains six heading levels.

```
1 <h1>My main title</h1>
2 <h2>My top level heading</h2>
3 <h3>My subheading</h3>
4 <h4>My sub-subheading</h4>
```

MONASH University

HTML Entity References

An HTML entity is a piece of text that begins with an ampersand and ends with a semicolon.

Entities are frequently used to display reserved characters (which would otherwise be interpreted as HTML code)

You can also use them in place of other characters that are difficulty to type with a standard keyboard.

MONASH University

HTML comment

Comments

As with most programming languages, there is a mechanism to write comments in the code.

- Comments are ignored by the browser and not seen by the user.
- It allows you to include comments in the code to say how your code works.
- Comments are very useful.

To turn a section into a comment, you will need to wrap them into the special markers `<!--` and `-->`

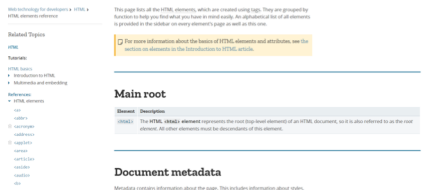
```
1 <p>I am not inside a comment.</p>
2 <!-- <p>I am inside a comment.</p> -->
```

Most IDEs have a shortcut to comment out blocks of code. Using features in an IDE is a good tool in your toolkit of skills.

MONASH University

More HTML elements

- What we have covered so far is only a few HTML elements, however there are many more than that.
- The entire list of HTML elements can be found here. Using this reference will be a skill that you will gain over time.



MONASH University

HTML5

HTML5 is the latest HTML5 standard.

Visual Studio 2017 which we will be using in this subject will create default project with HTML5 features.

- It represents two different concepts
- It is the new version of the language HTML with **new elements, attributes** and **behaviors**.
- It **also** has **a larger set of technologies** that allows the building of more diverse and powerful websites and applications.

HTML5 technologies can be broken down into different groups

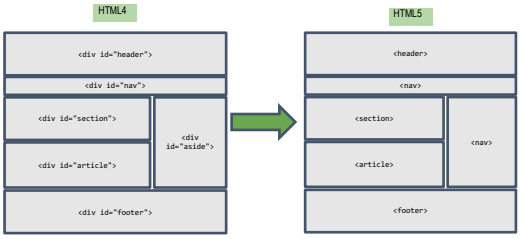
- Semantics - This allows you to describe more precisely what your content is.
- Connectivity
- Offline & Storage
- Multimedia
- 2D/3D graphics
- Performance & Integration
- Device Access
- Styling

MONASH University

HTML5 Semantic Elements

HTML5 introduced semantic elements.

It brings the ability to describe the structure of a web document with ***standard semantics.***



Obsolete and Deprecated Elements

These are old HTML elements which are **deprecated** and should **not** be used. You should not use them in newer projects and **should be replaced** in old projects as soon as possible.

These are some examples of them

- `<applet>`
- `<basefont>`
- `<bgsound>`
- `<big>`
- `<center>`
- `<command>`
- `<content>`
- `<font>`
- `<frame>`
- `<image>`

The complete list can be found here.

Schema.org

- Schema.org provides a collection of shared vocabularies webmasters can use to mark up their pages in ways that can be understood by the major search engines: Google, Microsoft, Yandex and Yahoo!
- https://schema.org/docs/gs.html

Relevant XKCD

Do you see any issues there?

XKCD is a webcomics which subject matter varies from statements of life and love to mathematical, programming and scientific jokes.

In HTML, all elements except self-closing elements like `<img>` should consist of an open and close tag of the same type.

HTML has also never been case sensitive, the practice of using uppercase tags for readability is long outmoded, and the mixing of cases in this example would be annoying to a developer.

HTML should also be properly nested. Any element whose open tag occurs inside a `<div>` must be closed before a `<div>` is closed.

The webcomic is meant to be a joke telling how to annoy a web developer. However, in real life you do often see bad markups like these in production settings.

Styling the Web

CSS – **Cascading Style Sheets**

What is CSS?

- The code you use to style your web page.
- Was released in 1996.
- CSS is **NOT** a programming language. It is also **NOT** a markup language. It is a **style sheet language**.
- It allows you to apply styles **selectively to elements** in HTML documents.
- In short, CSS describes how elements should be rendered on screen.
- CSS is standardized across browsers according to the specification by the w3c.
- Majority of modern web browsers implement **CSS3**

CSS continued….

- CSS is a language for specifying how documents are presented to users - how they are styled, laid out, etc.
- A document is usually a text file structured using a markup language.
- Presenting a document to a user means converting it into a usable form for your audience. Browsers like Firefox, Chrome or Internet Explorer, are designed to present documents visually, for example, on a computer screen.
- Web browsers apply CSS rules to a document to affect how they are displayed.

Anatomy of a CSS rule set

CSS - Cascading Style Sheet

```
p {
  color: red;
}
```

- The whole structure is called a rule set.
- Selector - The HTML element name at the start of the rule set. It selects the element(s) to be styled. **Selectors are used to target the HTML elements on our web pages that we want to style.**
- Declaration - A single rule like color: red; specifying which of the elements properties you want to style.
- Properties - Ways in which you can style a given HTML element. In the example, color is a property of the `<p>` element.
- Property value - Specifies the property value, in the example, the color is set to red.

MONASH University

Anatomy of a CSS rule set

CSS - Cascading Style Sheet

Other important parts of the syntax:

- Each rule set (apart from the selector) must be wrapped in curly braces `{ }`
- Within each declaration, you must use a colon `:` to separate the property from its values.
- Within each rule set, you must use a semicolon `;` to separate each declaration from the next one.

```
p {
  color: red;
  width: 10rem;
  border: 1px solid black;
}
```

MONASH University

Online Source Code Playground

JSFiddle also provides the ability to play around with CSS.

MONASH University

Different types of selectors

There are many types of selectors. It is also possible to make more specific selectors. Here are some examples of more common types of selectors.

Selector name	What does it select
Element selector	All HTML elements of the specific type
ID Selector	The element on the page with the specified ID (on a given HTML page, you are only allowed one element per ID) - ID is unique.
Class Selector	The elements(s) on the page with the specific class (multiple class instances can appear on a page)
Attribute Selector	The element(s) on the page with the specified attribute.
Pseudo-class selector	The specified element(s), but only when in the specified state, e.g. being hovered over.
Combinators	A combination
Multiple selectors	The idea here is that you can put multiple selectors on the same CSS rule, separated by commas, to apply a single

MONASH University

Padding, border, margin?

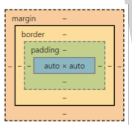
One important concept about CSS is that elements on your page can be thought of boxes sitting on top of each other.

CSS principle is based on the ***box model***. Each of these boxes has these properties.

Margin → The space around the outside of the element.

Border → The solid line that sits just outside the padding.

Padding → The space around the outside of the element.



MONASH University

Casing and spacing

- It is very important to keep casing and spacing consistent.
- Directories and files should be named completely in **lowercase** with **no spaces**.
- Reason being, many computers are case-sensitive, so there is a need to keep the naming conventions in such a manner.
- Browsers, web servers and programming languages do not handle spaces consistently.

In short…

- It is a good idea to write your file and directory names with lower cases with no spaces and with words separated with dashes.

MONASH University

Our Code Guide (Coding Convention)

- FIT5032 - Internet Application Development follows the Code Guide by Mark Otto.
- The Code Guide is a project for documenting standards for developing flexible, durable and sustainable HTML and CSS.
- The Code Guide here only applies to HTML and CSS.
- It can be found here.

Snippets from the Code Guide

- *Every line of code should appear to be written by a single person, no matter the number of contributors.*

References

https://developer.mozilla.org/en-US/docs/Learn/Getting_started_with_the_web/CSS_basics

https://html.spec.whatwg.org/dev/introduction.html#is-this-html5?