

# Higher Computing Science

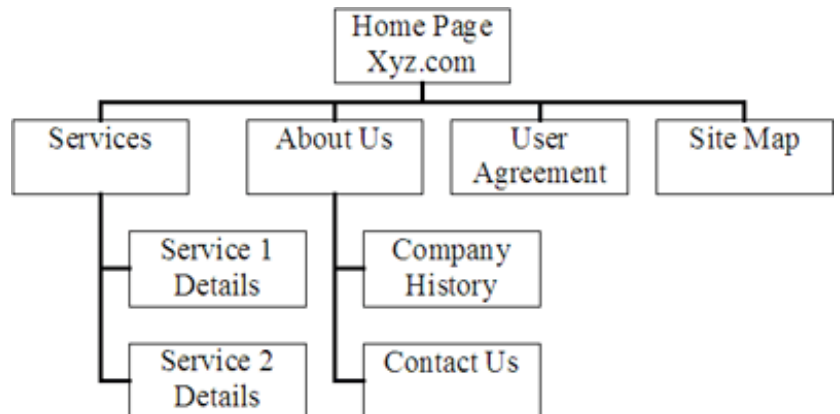
## Web design

### Summary notes

#### Design notations

The structure of a website can be planned using a **structure diagram**.

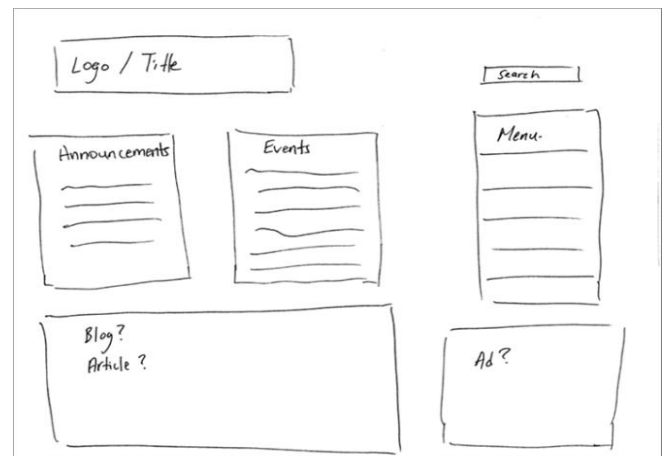
This will show how the pages in the site link together, usually in a hierarchical structure. This is an example of a **multi-level navigation** structure.



The layout of individual pages can be planned using **wireframes**.

These will show how text, graphics, links, etc will fit together on the page.

You may also consider noting formatting information, such as colours and fonts to be used.



#### Multi-level structure

- A multi-level navigation structure involves sub-levels of navigation.
- It can be useful for keeping websites with a lot of information organised.

For example, the BBC website, allows you to choose a category, then another category within that, then another category within that, and so on.



# HTML (HyperText Markup Language)

HTML is a language used to define the content of a webpage, using a variety of tags.

- An HTML page begins with the **<head>** element. This can include a variety of other elements which give information about the page, such as the title, meta information, location of style sheets, etc.
- After the **<head>** comes the **<body>**. The **<body>** element contains the content to be displayed.

Some more advanced HTML tags you should know:

Tag	Notes	Example
<code>&lt;title&gt;</code>	<ul style="list-style-type: none"><li>• Included in the <code>&lt;head&gt;</code> element of the page.</li><li>• Defines a title in the browser toolbar</li><li>• Displays a title for the page in search engine results</li><li>• Provides a title for the page when it is added to favourites</li></ul>	<pre>&lt;title&gt; ACME Corporation home page &lt;/title&gt;</pre>
<code>&lt;meta&gt;</code>	<ul style="list-style-type: none"><li>• Included in the <code>&lt;head&gt;</code> element of the page.</li><li>• Provides information about the page, not visible in the browser (i.e. information about information!).</li><li>• There is no ending <code>&lt;/meta&gt;</code> tag as it does not need to be closed.</li><li>• Properties include name, content and character set.</li></ul>	<p><b>Example 1</b></p> <pre>&lt;meta name="description" content="ACME Corporation - Unlikely contraptions for all your needs"&gt;</pre> <p><b>Example 2</b></p> <pre>&lt;meta name="keywords" content="ACME, inventions, traps, contraptions"&gt;</pre>
<code>&lt;style&gt;</code>	<ul style="list-style-type: none"><li>• Allows CSS (see below) styles to be defined within an HTML document</li><li>• Included in the <code>&lt;head&gt;</code> element of the page.</li></ul>	<pre>&lt;style&gt; p { margin-left:20px; } body { background- image:url("background.gif"); } &lt;/style&gt;</pre>
<code>&lt;script&gt;</code>	<ul style="list-style-type: none"><li>• Used to define a client-side script (see below), such as JavaScript.</li><li>• Alternatively, can provide a link to an external script file.</li></ul>	<p><b>Example 1</b></p> <pre>&lt;script&gt; document.write("Hello World!") &lt;/script&gt;</pre> <p><b>Example 2</b></p> <pre>&lt;script src="js/lightbox.js"&gt; &lt;/script&gt;</pre>
<code>&lt;button&gt;</code>	<ul style="list-style-type: none"><li>• Used to create a clickable button</li></ul>	<pre>&lt;button type="button"&gt;Click Me!&lt;/button&gt;</pre>

## CSS (Cascading Style Sheets)

CSS is a language used to define the look and formatting of HTML documents.

- CSS can be included within an HTML document, but it is preferable to link the HTML to a separate CSS file.
- Using a separate CSS to define the formatting of HTML documents can help to give a consistent look across a website, save time and create more efficient code. For example this allows multiple web pages to be linked and follow the same rules.
- The look of all pages in a site can be changed by editing only the CSS file.
- CSS can be used to provide different style rules depending on the type of device being used to view the webpage.

### Example

Selector

```
body
{
background-color:#d0e4fe;
}

h1
{
color:orange;
font-weight:bold;
}

p
{
font-family:"Times New Roman";
font-size:20px;
text-align:center;
}
```

Property

Value

This CSS sets the background colour for the body of the page, the colour and style of any text within <h1> tags, and the font and size of any text within <p>.

Any HTML file which is to use this formatting would simply include a line like this in the <head> section:

```
<link rel="stylesheet"
type="text/css" href="mystyle.css">
```

Where "mystyle.css" is replaced with the name of the CSS file.

## Inline rules (CSS)

An inline rule will apply CSS rules to a single HTML element. The styling rules are contained within the HTML tag.

```
<p style="color:red;">Red text</p>
```

This is generally an inefficient method of styling, as every element will require a separate rule, leading to **duplication** and excess code.

## Internal stylesheet rules (CSS)

An internal style sheet is where all the CSS rules are contained at the start of the same document where the styles are being applied. In the case of an HTML document this would be within the **head** section.

```
<head>
<style>
  h1{
    text-align:center;
  }
</style>
<h1>Welcome to XCSSIVE STYLIN</h1>
```

This is generally a better method of styling, as a single rule can apply to multiple elements of the same type, which reduces **duplication** and excess code. However, each document will require its own stylesheet which makes it difficult to make changes to styling that are consistent across multiple webpages.

## External stylesheet rules (CSS)

An external style sheet is where all the CSS rules are contained in an external document and these can be applied to multiple pages at once. The link to this is established using the link HTML element which is placed within the **head** section.

### HTML file

```
<head>
<link rel="stylesheet" type="text/css" href="styles.css">
</head>
```

```
<h2>XTERNAL CSS? It's no joke!</h2>
```

### CSS file

```
h2{
font-family:Verdana;
font-size:32px
}
```

This is generally the most convenient method of styling, as all rules are contained within one document and can be easily modified. Multiple files can be linked which ensures there is no **duplication** of code and that styling is **consistent** across pages.

## Classes and IDs (CSS)

Rules can be applied to HTML elements using appropriate selectors (e.g. p, h3, a) but IDs and classes can be used to be more selective.

### Class

A class is generally used for rules that will apply to several elements of the same type. An example could be a series of paragraphs that should appear as quotes (e.g. *in italics*).

### CSS

```
.quote{  
font-style:italic  
}
```

Selector (.)

### HTML

```
<p class="quote">To C or not to C is a programmer's question!</p>
```

### ID

An ID is generally used for rules that will apply to a single unique element which will not share rules with another. An example could be a greeting message that will only appear in the document once.

### CSS

```
#greeting{  
font-family:"Comic Sans";  
color:green  
}
```

Selector (#)

### HTML

```
<p id="greeting">Welcome to my How To Make Lasagne page! It's Certainly  
Considered Special</p>
```

## Conflicting CSS rules and priorities

There may be instances where a CSS rule property conflicts with another. In this case the language follows a priority and will apply the rule that is most suitable for the element. The order of priority is:

Inline -----> ID selector -----> Class -----> Type selector

Overrides

Overrides

Overrides

## Static webpages

- A **static webpage** looks exactly the same every time it is accessed, the content does not change.

## Dynamic webpages

- **Dynamic webpages** can contain different content depending on the conditions in which they are accessed, meaning that users can be presented with a page relating specifically to their needs.
- **Database-driven** pages are the most common example of dynamic pages. If information changes in a database which the page accesses, then the content of the page will change to reflect this. For example, online shopping, banking, buying tickets, etc.

## Interactive webpages

- **Interactive webpages** offer a more immersive experience, allowing the user's actions to affect the content. Creating interactive pages usually involves the use of JavaScript or Flash.

## Scripting

- Including sections of program code in your website, to help make it more dynamic or interactive, is called scripting.

### Client-side scripting

- A section of code is sent to the client computer and executed there, in the browser.
- It is commonly used to add interactive elements to a webpage.
- JavaScript is a language commonly used for client-side scripting.

### Server-side scripting

- A section of code is executed by the web server.
- It is commonly used to create dynamic webpages, often connected to a database on the server.
- Popular languages for server-side scripting include PHP, ASP, Perl and ColdFusion.

## JavaScript

JavaScript is a scripting language used to control the behaviour of webpages.

- It can be used to change HTML elements. E.g. Change an image when it is clicked.
- It can be used to change CSS styles. E.g. Change the colour of the page when a button is clicked.
- It can be used to validate user input. E.g. Asking the user to create a password and checking if it is valid.
- Displaying changes when the user moves the mouse cursor over an element.

- Scripts can be included in the <head> element of a page, then referred to at any point in the <body>
- Alternatively, the entire script can be included at the appropriate point in the body.
- Where a script is used many times throughout a website it is most efficient to put it in an external file, then link each page to it using the <script> tag.

## Examples

<pre>if (time &lt; 20) {     greeting = "Good day"; } else {     greeting = "Good evening"; }</pre>	<p>This script displays a message depending on the hour of the current time.</p>
<pre>document.getElementById("demo").innerHTML = "Hello Dolly.";</pre>	<p>This script changes the content of an HTML element with the ID "demo".</p>

## Mouse events

Using JavaScript, functions are created that will change what appears on screen depending upon what the user does with the mouse.

This script has been written to create two functions, mouseOver() and mouseOut().

```
<script>
function mouseOver() {
    document.getElementById("select").style.color = "green";
}
function mouseOut() {
    document.getElementById("select").style.color = "black";
}
</script>
```

The first 3 lines show a function called mouseOver(). This function will change the style of an element to green if the ID of that element is called 'select'.

The following 3 lines show another function called mouseOut(). This function will change the style of an element to black if the ID of that element is called 'select'. For this code to run, it is necessary for there to be an HTML element that references the ID 'select'.

In the HTML shown below a heading is being used. This heading is identified as 'select'.


```
<h1 id="select" onmouseover="mouseOver()"
onmouseout="mouseOut()">Products</h1>
```

When the mouse is moved over the element the onmouseover event is triggered. This calls the mouseOver function that changes the colour property of the HTML text to green.

Products ←



When the mouse is moved away from the element the onmouseout event is triggered. This causes the mouseOut function to change the colour of the heading element back to black.

Products → 

## PHP (PHP Hypertext Pre-processor)

PHP is a server-side scripting language used to create dynamic and database-driven webpages (in conjunction with SQL).

- A PHP script is executed on the server but any output can be returned to the browser in a variety of forms, such as HTML, images or PDF.
- To use PHP in your webpages, you must find web hosting which supports PHP and MySQL.
- PHP files can contain HTML, CSS and JavaScript, as well as PHP.
- PHP scripts begin with <?php and end with ?>

### Examples

<pre>&lt;?php \$t=date("H");  if (\$t&lt;"20") {     echo "Have a good day!"; } ?&gt;</pre>	<p>This script displays a message if the hour of the current time is less than 20.</p>
<pre>&lt;?php setcookie("user", "Alex Porter", time()+3600); ?&gt;</pre>	<p>This script saves a cookie to the user's computer.</p>

## SQL (Structured Query Language)

SQL is a language used to access and manipulate databases.

- It allows you to add/delete/amend records and fields, search records and sort records.
- It is used in conjunction with a server-side language, like PHP, to create database-driven websites.

### Examples

<pre>SELECT CustomerName, City FROM Customers;</pre>	<p>Shows only the CustomerName and City fields from the Customers table.</p>
<pre>SELECT * FROM Customers WHERE Country='Mexico';</pre>	<p>Shows only the records from the Customers table which have "Mexico" in the Country field.</p>
<pre>SELECT * FROM Customers ORDER BY Country ASC;</pre>	<p>Shows all the records from the Customers table in ascending order by Country.</p>



## Database driven websites

Database driven websites also allow users to fill in forms that can then be sent as queries to the database, allowing for real-time feedback on things like the number of tickets available for a concert or the number of seats left on flight.

If the information stored in the database changes, the web page connected to the database will update automatically. For example, online banking sites where customers can log in and check their bank account balance are updated in real time.

The main advantages of dynamic database driven websites are:

- real time update of webpages when the content of a server-side database changes
- user specific/targeted content
- possibility of e-commerce
- real time update of content linked to other database driven sites or servers, eg weather widgets
- ability to gain feedback or queries from the user via forms.

### Retrieving information from a database using a query

Server-side web languages can be used with a database driven website to find and display relevant information. For example, how many items are currently in stock for a product on an online webstore.



The screenshot shows a product page for a Gigabyte GeForce GTX 1060 Windforce OC 6GB GDDR5 Graphics Card. On the left is an image of the product box and the card itself. The main content area includes the brand name 'GIGABYTE', the product name, a star rating of 4.5 stars with 208 reviews, and a list of specifications: NVIDIA GeForce GTX 1060 GPU, Gigabyte Windforce OC Edition, 6GB GDDR5 Memory, and PCI-Express x16 Interface. There are buttons for 'View full description' and 'View add-ons and accessories'. On the right, the price is listed as £289.98 (inc. vat), with a crossed-out price of £360.00 and a saving of £80.01. A 'FREE DELIVERY' badge is present. A green 'Add to Basket' button is prominent. Below this, there are 'Add-on Products' including Intel Core i5 6600 3.3G... for £189.99 and Intel Core i7-6700K 4G... for £293.99. At the bottom right, it states 'In Stock' and 'Order within 6 hours for delivery on Tuesday, 8th May'.

Steps taken to generate webpage using query:

- Input is received from the user (mouse click, form data etc.) and this may be **validated** to ensure that it meets a certain criterion.
- Data captured from the webpage/webform is used to construct a query for the database.
- A connection with server/database established using a server-side web language (e.g. PHP/Perl etc.)
- The query is run on the server using a suitable language (e.g. SQL) and the results of this are generated.
- The result of the query is processed and used to construct HTML that will display the information on the webpage.

## Compression

Appropriate use of compression can reduce the load times of webpages. The webpage itself can be compressed, reducing the amount of data which has to be sent to the client computer. The browser will decompress the page before displaying it.

### Lossless compression

Lossless compression reduces file sizes without removing any data. Quality is not affected.

PNG	Graphics	24 bit colour, supports transparency but not animation
GIF	Graphics	Restricted to 8 bit colour, supports transparency and animation. LZW compression.
FLAC	Audio	Free Lossless Audio Codec. Retains full quality of original and can reduce size to about 50-60%.

### Lossy compression

Lossy compression reduces file sizes by removing some data. Quality is reduced.

JPEG	Graphics	24 bit colour, adjustable compression level, DCT compression
MP3	Audio	Perceptual coding <ul style="list-style-type: none"><li>• Removes sounds outwith human hearing</li><li>• Removes quieter sounds masked by louder sounds</li><li>• Bass frequencies are converted to mono</li></ul>
MP4	Video	Stores key frames and only the changes in between them. Interframe compression.

## SEO (Search Engine Optimisation)

SEO is the process of improving the ranking of your webpage in search engine results, in order to increase traffic.

Some SEO techniques include:

- Making sure each page of your site has a unique, concise and descriptive title contained in the <title> tag.
- Using the description <meta> tag to include a summary of each page.
- Using concise URLs.
- Submit an XML sitemap to the companies which run the search engines
- Using brief, descriptive filenames and alt text for images

## Search engines and web crawlers

- A web crawler (aka spider) is a piece of software which follows links from web page to web page, analyses the text on each page, and indexes the pages with related keywords.
- Search engines then use this index to find relevant pages when you carry out a search.

## Optimisation of code

Load times of webpages affect traffic. By making code more efficient, file sizes, and therefore load times, can be reduced.

Some techniques include:

- Using a single CSS file to hold all the formatting information for the site.
- Putting all JavaScript into a single external file, to reduce fetches from the server.
- Remove/reduce internal commentary and white space.

## Accessibility

Web accessibility means making sure that all users can easily access and navigate your website.

Some ways in which accessibility can be improved include:

- Using CSS media queries to identify the type of device accessing the webpage, and adjust display accordingly.
- Using appropriate alt text for images, as these can be read by reader software.
- Make all functionality accessible from the keyboard.
- Make all text content easily readable and understandable.

## Usability

Usability refers to how easy your website is to use.

Some ways in which usability can be improved include:

- Making main navigation easily identifiable, with easily identifiable links.
- Using consistent style throughout.
- Ensuring the most important content is above the "fold" (the point where the bottom of the screen cuts off the page).
- Including a clear path to contact details and information about the organisation.

## Beta testing

Beta testers carry out testing on the final website before it goes live.

Beta testers could be anybody not involved in the development of the site, such as potential customers.

The purpose of beta testing is to identify any problems that occur in normal use.

## Usability testing

Usability testing involves testing how easy the website is to use.

### Eye-tracking

- Eye tracking software and hardware allows testers to analyse the way in which a user reads the information on a webpage.
- This can help in adjusting the layout to focus attention on the most important information.

### Think aloud

- Think aloud testing involves users describing their thought process as they try to complete a task on a website.
- This can help in identifying design issues which may cause problems for users.