

Assessment Report

Student name: Mark Beresford

Student Number: 471731221

Unit/Cluster: ICTWEB519/520 Web design cluster

Assessment Number: 2



Contents

[Title 2](#_Toc133227678)

[1. Project Requirements (Part 1) 2](#_Toc133227679)

[Client Requirements 2](#_Toc133227680)

[Wireframes 5](#_Toc133227681)

[Styling Information 7](#_Toc133227682)

[Typography 7](#_Toc133227683)

[Iconography 7](#_Toc133227684)

[Colour Scheme 8](#_Toc133227685)

[Markup Language Comparison 8](#_Toc133227686)

[2. Test, debug and validate (Part 3) 9](#_Toc133227687)

[Validation 9](#_Toc133227688)

[Accessibility Testing 14](#_Toc133227689)

[Browser/Device Testing 16](#_Toc133227690)

[Cyber security measures 27](#_Toc133227691)

[Authentication process 30](#_Toc133227692)

[Programmatically engineered solutions to avoid cyber-attacks 30](#_Toc133227693)

[Internet protocols 31](#_Toc133227694)

[Project Signoff 31](#_Toc133227695)

[3. Knowledge Questions (Part 4) 32](#_Toc133227696)

[Appendices 35](#_Toc133227697)

[References 35](#_Toc133227698)

Title

ICTWEB519/520 IT Assessment 2 Report

Date: 11/05/2023

1. Project Requirements (Part 1)

Client Requirements

* 1. The client’s aim is to use its local reputation to increase its customer base of young families with children, and aid in the diagnosis of a range of eye conditions to help the overall quality of life in the local community.
  2. The main target audience for the page is local families with young children, but any local requiring an optometrist will find their services useful.

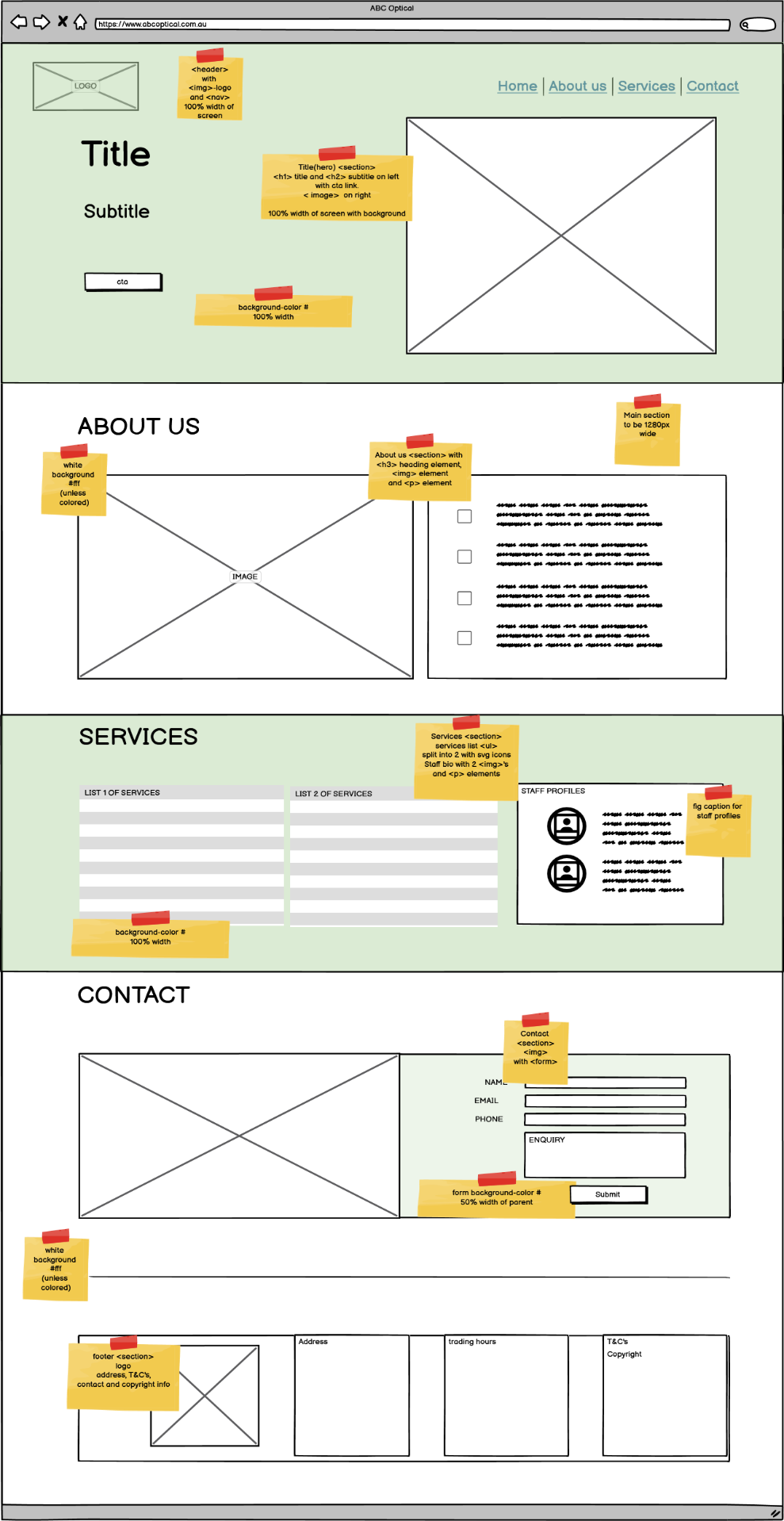
1.3

|  |  |  |  |
| --- | --- | --- | --- |
| LEGAL OBLIGATION | | ACTIONS REQUIRED | SOURCE |
| Privacy Act | | * Inform your customers of what measures you have in place to protect their personal details, such as names, addresses and credit card details. * Check your privacy policy is compliant with Australian privacy law and is easily accessible on your website. | <https://www.business.qld.gov.au/running-business/digital-business/online-risk-security/legal-obligations> |
| Legal Obligations and jurisdictions | * Assess your online business activities and the geographic areas your business operates within and identify the legal obligations that apply. * Create a plan to ensure that all your business activities comply. * Conduct regular reviews or sign up for updates to make you aware of any changes, or seek professional advice. | | <https://www.business.qld.gov.au/running-business/digital-business/online-risk-security/legal-obligations> |
| Email Marketing | * Comply with the [Spam Act 2003(Cwlth)](https://www.legislation.gov.au/Series/C2004A01214) | | <https://www.legislation.gov.au/Details/C2016C00614> |
| Copyright | * Images and content to meet copyright requirements * Have permissions to use intellectual property | |  |
| Consumer Law | * Online advertising is not false or misleading. * Online reviews are genuine. * Uphold responsibilities to customers. * Comply with Australian Product/Mandatory safety laws | | <https://www.business.qld.gov.au/running-business/digital-business/online-risk-security/legal-obligations> |
| Website term of use | * A terms of use document that includes information on protecting your business ideas, disclaimers and rules applying to anyone who interacts with your website. | | <https://www.business.qld.gov.au/running-business/digital-business/online-risk-security/legal-obligations> |
| Terms and Conditions | * If you sell products online, include terms and conditions outlining the operations of your business sales, payment options, cancellation policies, consumer guarantees, and return and refund policies. * You should also have a process for customer complaints. | | <https://www.business.qld.gov.au/running-business/digital-business/online-risk-security/legal-obligations> |
| Business Information | * Contact details are included and easy to find | | <https://www.business.qld.gov.au/running-business/digital-business/online-risk-security/legal-obligations> |
| Client Agreement | * If you provide services online, include a client agreement. * This should contain information about payment, expectations, termination of the agreement, and the process for handling disputes | | <https://www.business.qld.gov.au/running-business/digital-business/online-risk-security/legal-obligations> |
| Electronic Transactions | * Provide payment mechanisms that are easy to use and give appropriate security. * Clearly explain the security and authentication methods you use so customers can assess any risks | | <https://www.business.qld.gov.au/running-business/digital-business/online-risk-security/legal-obligations> |
| Accessibility Standards | * Comply with WCAG 2.0/2.1 standards * Ensure that WCAG 2.1 success criteria support mobile use-cases * Define a clear conformance model for WCAG 2.1/dot.x releases * Ensure the conformance structure utilizes the WCAG 2.0 A / AA / AAA model | | <https://www.w3.org/TR/WCAG20/>  <https://w3c.github.io/wcag21/requirements/> |

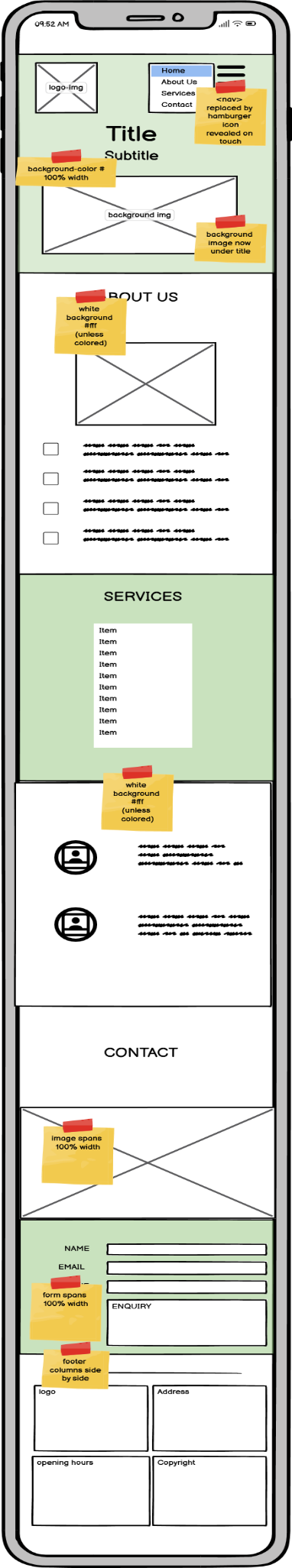
Wireframes

1.7

**PC/Desktop Version of Website**

**

**Mobile Phone Screen of website**



Styling Information

Typography

1.8

|  |  |
| --- | --- |
| Main Styles | |
| Title/Heading 1 | 54px |
| Heading 2 | 20px |
| Heading 3 | 25px |
| Link: link, active and hover | 20px |
| Paragraph | 16px/20px |
| Fig caption | 18px/14px/12px |
| Unordered Lists | 16px |
| Labels | 12px |
| Inputs | 16px |
| Buttons | 20px |
| Font-family | |
| Headings | Lato |
| Paragraphs/list | Montserrat |
| Back-up | Sans serif |

Iconography

The iconography format style I chose is from the heroicons website ([https://heroicons.com/)](https://heroicons.com/) as they are SVG format, and have a choice of outline, solid or mini in which I selected the outline style. These can then be coloured using stroke in CSS.

Below is listed the specific icons I chose and why:

|  |  |
| --- | --- |
|  | To represent the business is local |
|  | A caring approach to children |
|  | For Optometrist |
|  | Positive health and well-being |
|  | A check to signify a service provided |

Colour Scheme

|  |  |
| --- | --- |
| **COLOUR SCHEMES** | |
| **PAGE CONTENT – From Style Guide 1** | |
| **Background:**  Overall background will be white(#fff), the Title section, Services section and form background will use the colour palette green(#87d195  ). | **Text:**  Paragraphs will use a lighter black(#), while headings will use a darker version of the colour palette green(#). |
| **Buttons/Links:**  Button/link backgrounds will all share the same green colour(#216b2e) with box shadows.  Text will be the cream(#ffec99).  CTA buttons/links will use a paler red(#cd5b5b)  **Icons:**  Dark green icons, used on light green background where needed., | **FORMS:**  **Background:**  form background will use the colour palette green(#).  **Labels:**  Capitalize .  **Input fields:**  Green borders  **Submit Button:**  As Call to action button |
| **SCHEME – (From Style Guide 1)**  Colour Palette 3 |
| **White space** | Each section will occupy a height of 100vh |

Markup Language Comparison

1.9

|  |  |  |  |
| --- | --- | --- | --- |
| **SELECTION CRITERIA** | **HTML** | **XML** | **XHTML** |
| Browser compatibility | All major browsers | All major browsers | All major browsers |
| Tag type | Predefined | User Defined | User Defined |
| Case sensitive | No | Yes | Yes |
| Data storage | In tags | Can use a separate file | In Tags |
| File size | Small | Large | Small |
| **Justify your selection** *(minimum 30 words)***:**  With no case sensitivity and tags being predefined, HTML is an easier option for markup.  XHTML does not include all of the elements that are included in HTML5, so validation is error prone. File size with XML could be high for a single page website. | | | |

2. Test, debug and validate (Part 3)

Validation

3.3

First round of HTML Validation

File: index.html

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

W3C Markup Validator and FreeFormatter indicated an issue with the HTML having too many hyphens in the comments, which was caused by an extension I use in VS code, so I removed the extra hyphens.

W3C Markup Validator also found an issue with the favicon HTML code, having a forward slash in the closing of the element which link elements in the head do not need.

Second round of HTML Validation

A screenshot of a computer

Description automatically generated with medium confidence

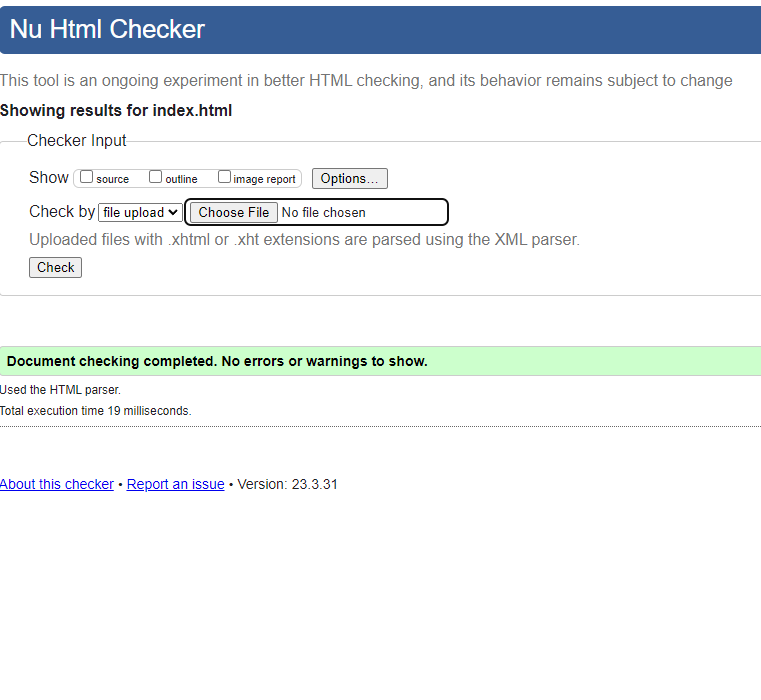
A screenshot of a computer

Description automatically generated

No issue with the HTML having too many hyphens in the comments anymore.

Favicon Issue remains, so removed forward slash from favicon link element.

Third round of HTML Validation



No Issues remain.

First round of CSS Validation

File: style.css

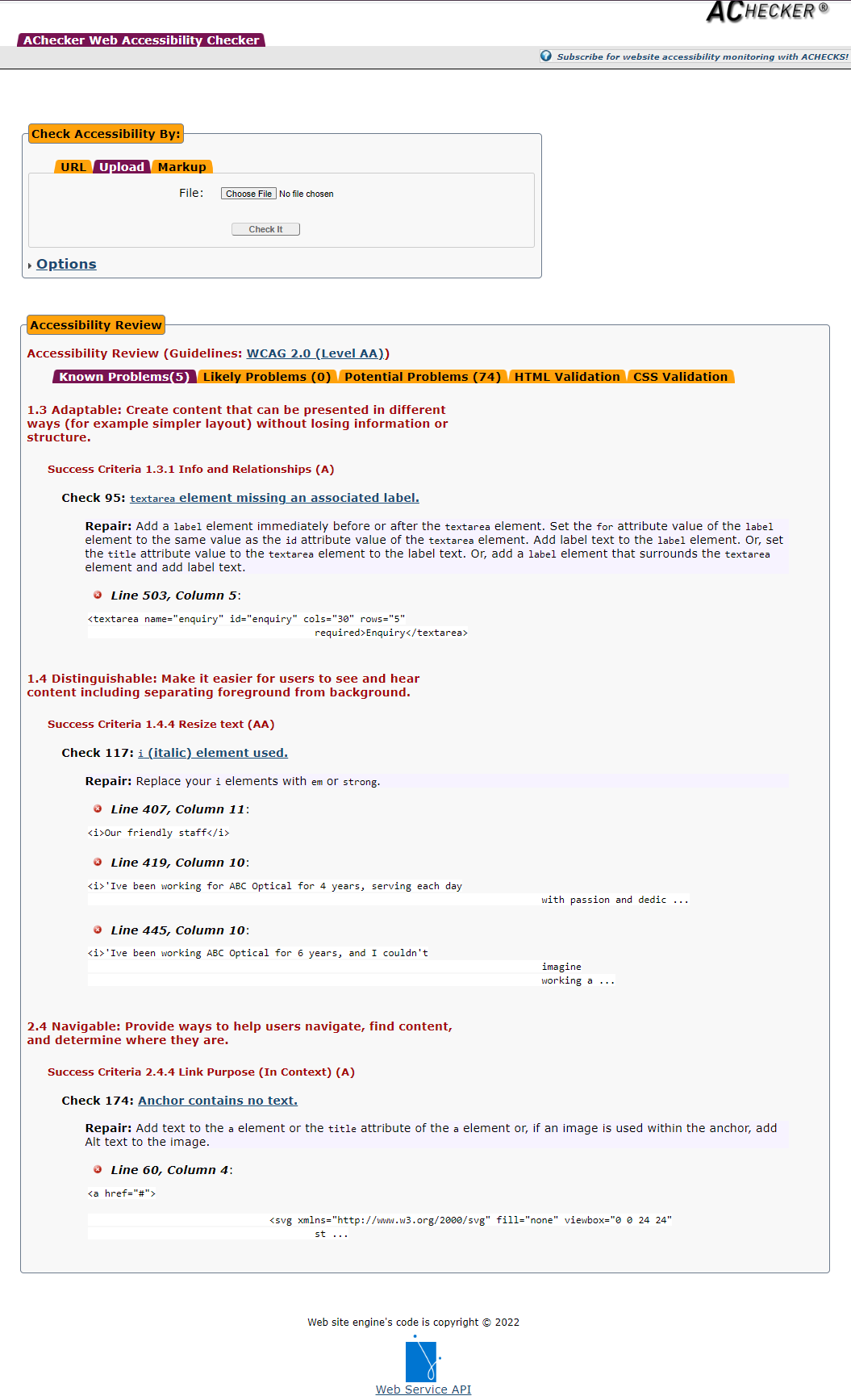
A screenshot of a computer

Description automatically generated with medium confidence

The CSS file passed the Validation on the first attempt.

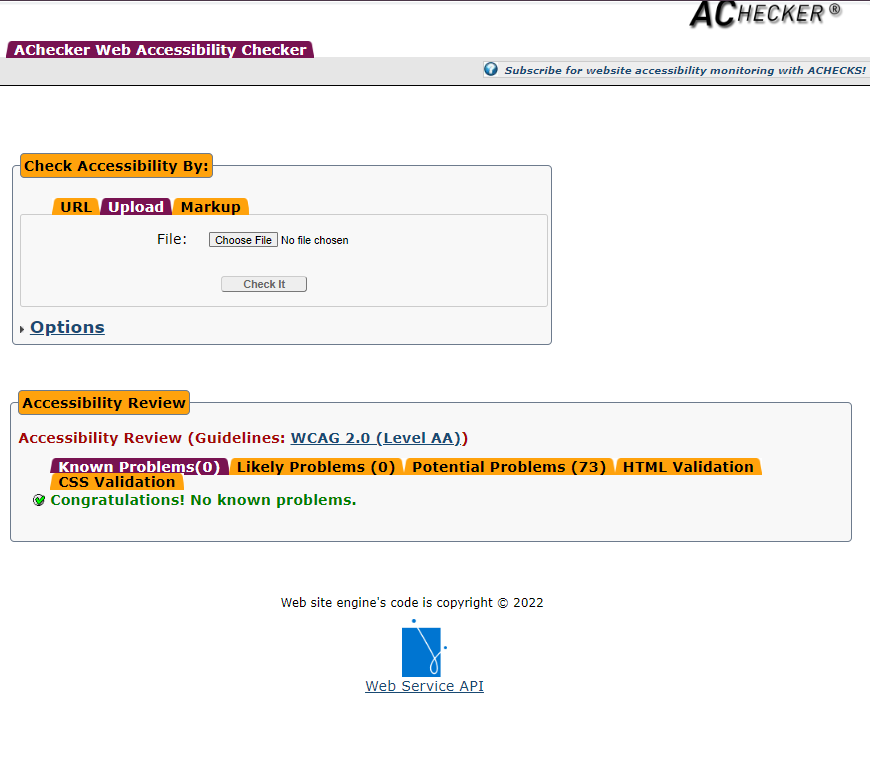
Accessibility Testing

3.4



5 problems identified on initial test:

* No label for textarea – added label.
* <i> used instead of <em> 3 times for italics – changed all <i> to <em>.
* <a> used for hamburger menu icon which holds no text – changed element to button.



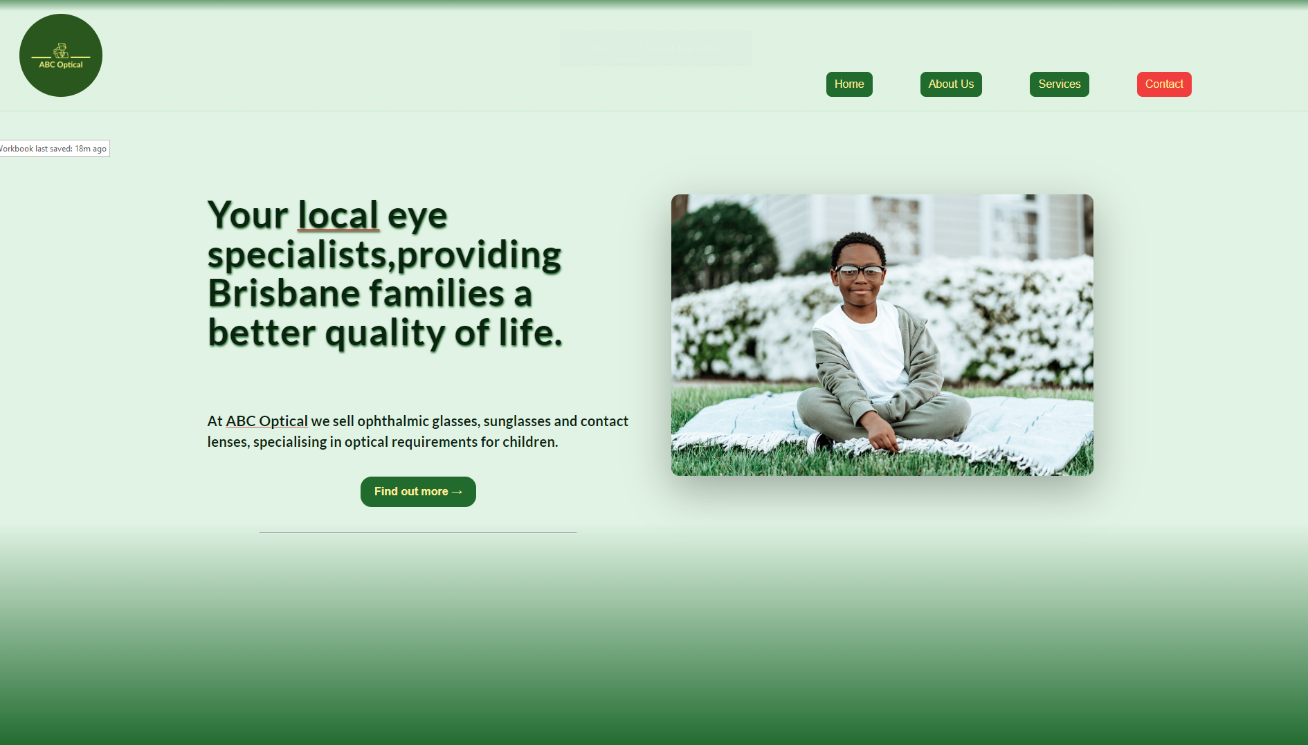
0 known problems.

0 likely problems.

75 potential problems – mainly suggestions around alt text for images.

Browser/Device Testing

3.5

The following screenshots are the desktop version of the website on Google Chrome V113.0.5672.63

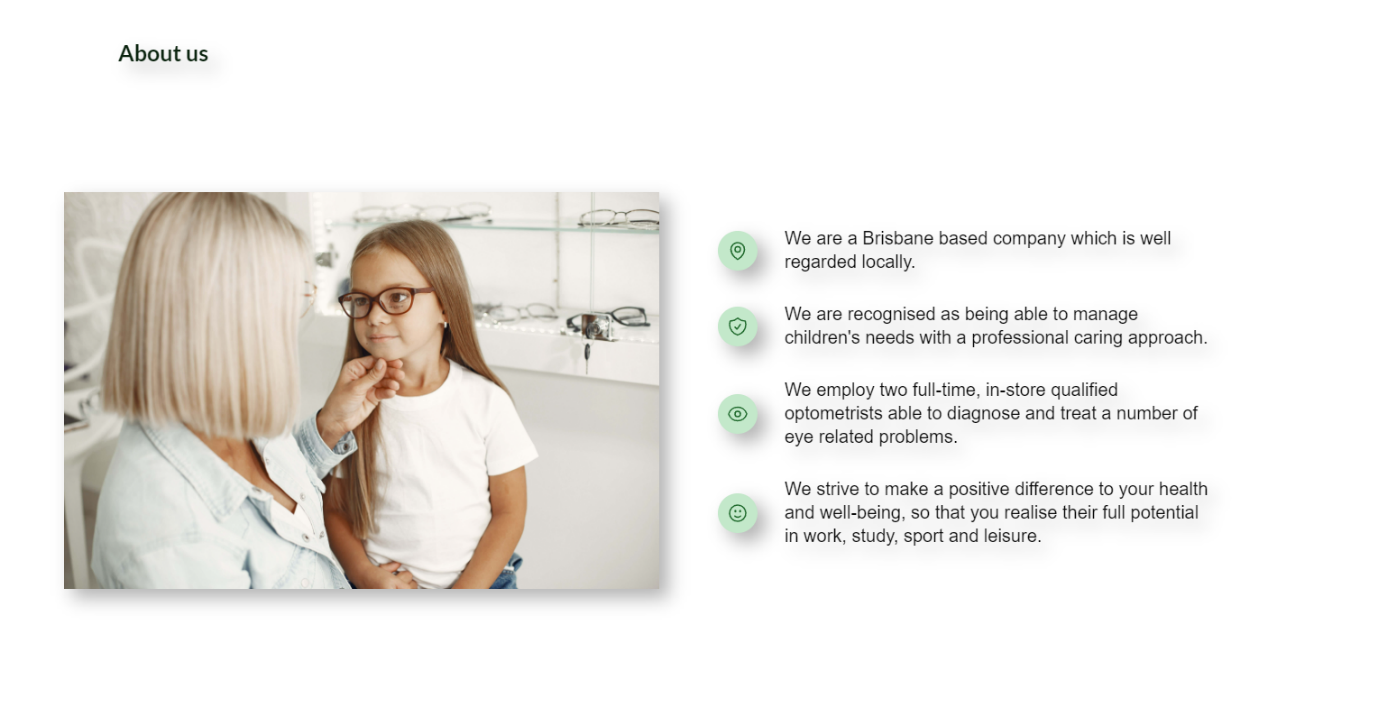
Figure 1 - Google Chrome V113.0.5672.63

Figure 2 - Google Chrome V113.0.5672.63

A screenshot of a computer

Description automatically generated with low confidence

Figure 3 - Google Chrome V113.0.5672.63



Figure 4 - Google Chrome V113.0.5672.63



Figure 5 - Google Chrome V113.0.5672.63

The following screenshots are the desktop version of the website on Opera Browser V:98.0.4759.15

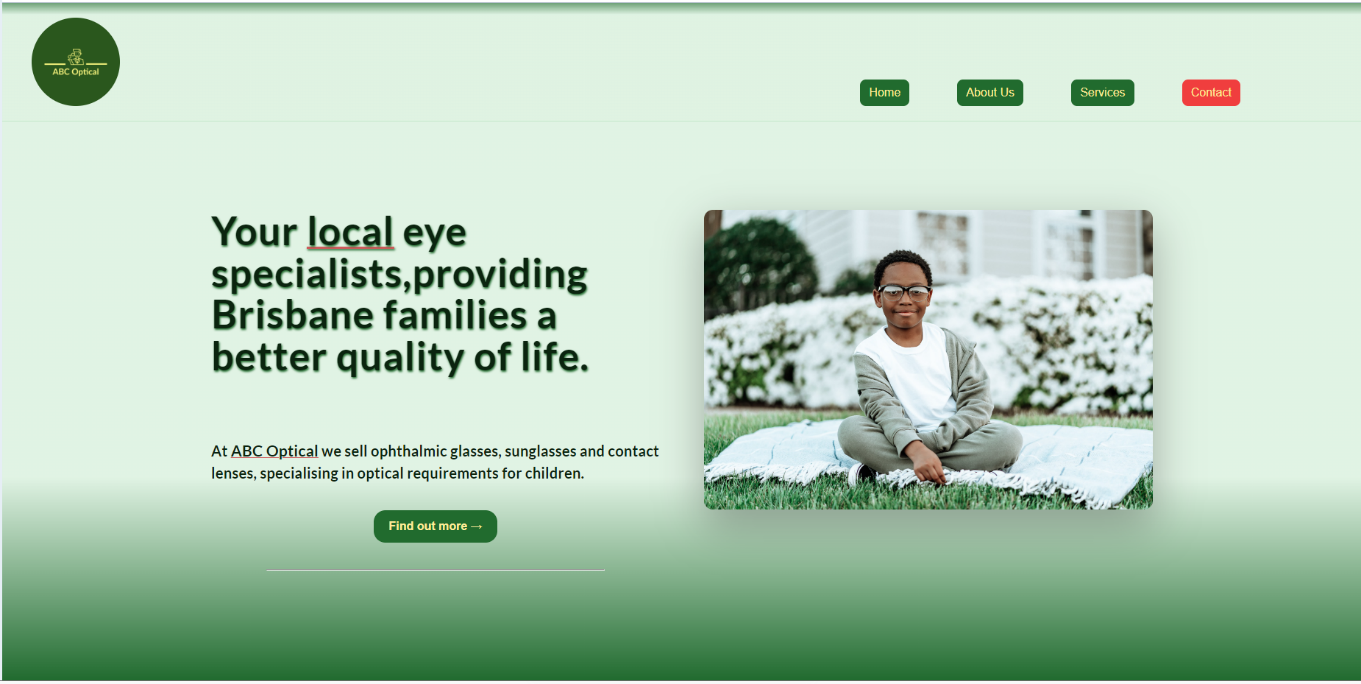


Figure 1 – Opera Browser V:98.0.4759.15



Figure 2 – Opera Browser V:98.0.4759.15

A picture containing text, screenshot, design

Description automatically generated

Figure 3 – Opera Browser V:98.0.4759.15

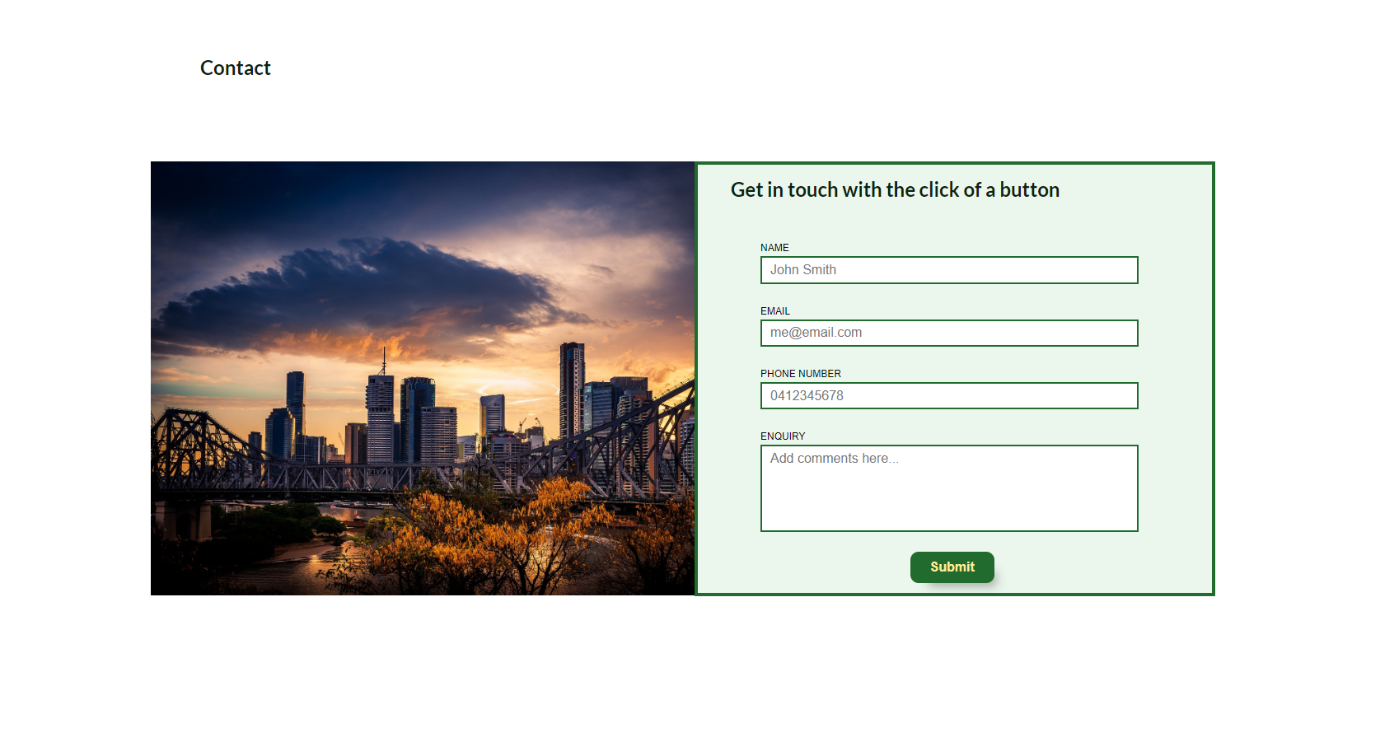


Figure 4 – Opera Browser V:98.0.4759.15

A close-up of a computer screen

Description automatically generated with low confidence

Figure 5– Opera Browser V:98.0.4759.15

The following screenshots are of the mobile phone version of the website using the

Samsung Galaxy S20 Ultra.

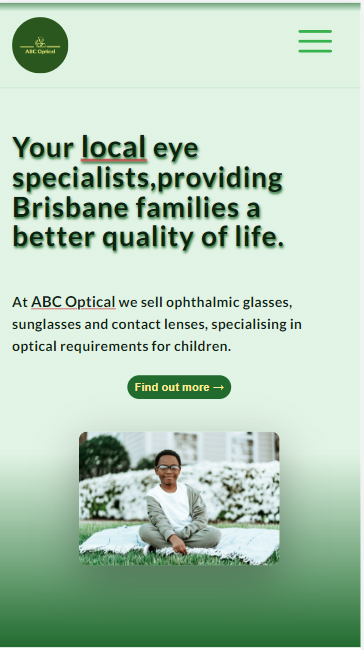


Figure 1 – Samsung Galaxy S20 Ultra

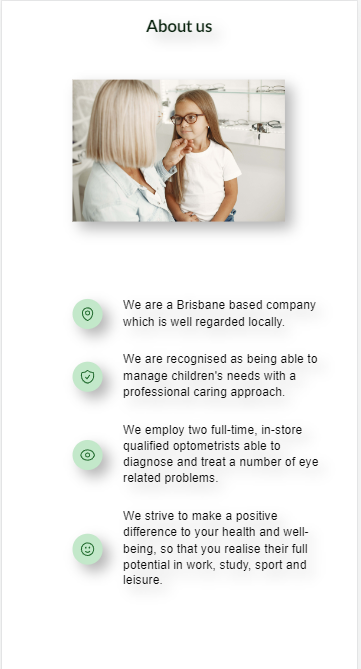
**

Figure 2 – Samsung Galaxy S20 Ultra

**

Figure 3 – Samsung Galaxy S20 Ultra

**

Figure 4 – Samsung Galaxy S20 Ultra

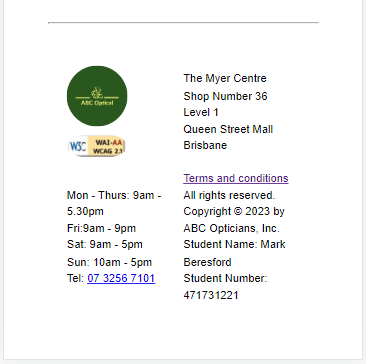
**

Figure 5 – Samsung Galaxy S20 Ultra

The following screenshots are of the mobile phone version using the iPhone 12 Pro:

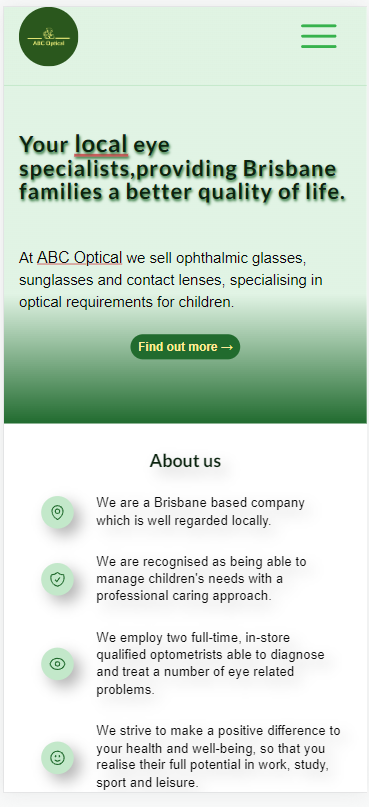


Figure 1 - iPhone 12 Pro

A screenshot of a phone

Description automatically generated with medium confidence

Figure 2 - iPhone 12 Pro

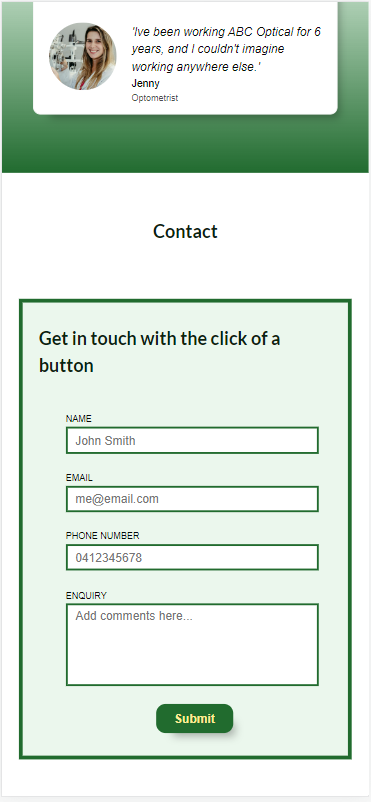


Figure 3 - iPhone 12 Pro

A screenshot of a phone

Description automatically generated with low confidence

Figure 4 - iPhone 12 Pro

Cyber security measures

3.6

Authentication process

a) Ways avoid or reduce cyber security attacks at the authentication process:

**Two-Factor Authentication**

Add an additional layer of protection to the user/password authentication process by adding 2FA (Two-factor authentication).

2FA verifies a user with two different approaches:

1. username/password
2. A phone or device (with an app generated code sent to the phone/device)

**Multi-Factor Authentication**

Multi-factor authentication is an authentication method that requires a user to verify their identity by providing more than one piece of information that identifies them(usually a fingerprint or secret security key/app generated code).

If someone was to obtain your username and password, they would still need your fingerprint or a code that has been sent to your phone.

<https://www.freecodecamp.org/news/user-authentication-methods-explained/>

Programmatically engineered solutions to avoid cyber-attacks.

b) Using programmatically engineered solutions to avoid cyber-attacks:

**Use Parameterised Prepared statements**

Parameterized prepared statements can be named and reused with different parameter values, so the only variation will be within the parameters, providing more security against SQL injections.

<https://jaredablon-31568.medium.com/how-to-prevent-sql-injection-vulnerabilities-how-prepared-statements-work-f492c369614f>

Internet protocols

c) Using internet protocols to prevent cyber security attacks:

**SSL Certificates**

An SSL certificate added to your website ensures a HTTPS (Hyper Text Transfer Protocol Secure) appears in the URL.

This process ensures that any data transmitted between the server and a web browser is secure and cannot be intercepted by third parties.

<https://medium.com/codex/what-is-ssl-for-dummies-eb26161c2b69>

Project Signoff

3.7

|  |  |
| --- | --- |
| **PROJECT SIGNOFF**  Signing off on this document signifies that the client representative is satisfied with the completed project. The client acknowledges satisfaction and completion with all elements of the project. | |
| Client Representative Signature:  Date: 24/05/2023 | Developer Signature:  Date: 24/05/2023 |
| **PROJECT NOT APPROVED**  Please provide feedback below and attach any relevant information to assist the project team in making the changes needed. | |

3. Knowledge Questions (Part 4)

* 1. 3 methods that could be used to apply CSS to web pages.
* Inline CSS – Applied to the HTML element on the markup page using the <style> attribute. As this is applied on the markup page, you don’t need to create and upload a separate document (Stylesheet). Inline styles have the highest specificity over any internal/external styles but can be time consuming and unorganized.
* Internal CSS – Created using the <style> attribute in the <head> of the HTML page, all styles are place inside here. This keeps your HTML and CSS code in one page, so you don’t need to upload multiple files and is more organized than inline styles. It also has the next highest level of specificity. This will however increase the size of the HTML page and the pages’ loading time.
* External CSS – Although this method has the least high specificity, it is by far the cleanest method as all styles are organized on a separate page, which can also be used by multiple HTML files. This provides an easily maintainable environment as adjustments to the CSS file can affect multiple pages. The drawback is the page needs to be loaded separately so the visual affects will not be applied until the page is loaded, which can also affect the download time of the page.
  1. Simplicity, visual hierarchy and grid-based layout are web design principles:

Simplicity:

Using easily understandable language engages the user and keeps them focused on the page as industry jargon and unnecessarily complicated words can be off putting.

Using images to convey meaning cuts down the clutter on the page and can be visually more appealing, keeping the users interest solely on the page.

Removing all unnecessary elements creates more focus to the main elements of the page (80/20 rule), and again cutting down on the clutter.

Use call-to-action link/ button to guide users easily take the next step of interaction.

A good navigation will make the page easy to use, and help users find the content they are seeking.

Visual Hierarchy:

Draw attention to the most important elements of the page and define a flow for users to follow.

Using larger, meaningful images will help to convey your message instead of too much text causing more clutter.

Use whitespace strategically to emphasize key elements and break up sections.

Use background colour to highlight the components of the page (e.g. Cards, forms, testimonials).

Grid-based layout:

Simple grid layouts can keep things tidy and easy to follow, using boxes and cards in columns and rows.

More complex designs like Z-patterns, F-patterns add more interest to the page and can be combined with simple layouts to break things up.

Single column pages with added components keep things simple and easy to follow, and also make responsive design easy to implement.

Multi column pages take design to the next level, and whilst can be hard to implement, increase the creative look of the page.

* 1. Evaluate two (2) web design principles used in front end web development .

**Mobile-first approach**

Web designers start product design with mobile devices first. The mobile layout and functionality are set as the default configuration, then additional information can be added on top. With the number of mobile users surpassing desktop users globally, a mobile-first design is an integral part of web development. Boostrap framework is ideal for a mobile-first approach, helping with drawbacks such as mobile navigation and control mechanisms.

**Navigation**

Keeping navigation simple and logical is a must if you want to keep your user engaging with your page. A streamlined easy to follow navigation help to increase the user experience, while confusing and unclear navigation makes the experience difficult and unattractive. You can also use navigation to point the user to the content you want them to see. Navigation can take many form, whether traditionally across the header, or down the sidebar, or even both for multiple navigations. Frameworks such as React, Angular and Vue provide many types of navigation to use.

* 1. Compare HTML and XHTML.

HTML and XHTML are closely related, with XHTML coming from the family of XML markup language.

One of main difference being the syntax used, with XHTML being strict around the opening and closing of tags, where as HTML allows some shortcuts especially with elements having optional tags, or elements without end tags. XHTML is also stricter with nesting, as all elements must be properly nested.

The 3 main differences between HTML and XHTML are

1. HTML is not case sensitive where XHTML is.
2. All XHTML syntax must be lower case, where it doesn’t with HTML
3. HTML tags are pre-defined, where as in XHTML you define your own tags.
   1. Present and outline at least three (3) advantages of creating and implementing organisational procedures to develop CSS.
4. Organising your CSS simplifies the maintenance. Finding and changing lines of code becomes easier, and also cuts down maintenance time.
5. Using an organised naming architecture (like BEM) makes your website scalable, so it is suited to any size website. You will not get conflicts in your code if you follow BEM guidelines and keeps your code cleaner. BEM also cuts down time if you use a CSS preproceesor like SASS/SCSS.
6. On many bigger projects, there will be a team of developers working on the same website, so keeping code readable and following a set of organised procedures will not slow things down or make things difficult, compromising the final product.
   1. CSS standard rules:
7. Spaces after function parameters. rgb(123, 123, 123);

Sometimes commas can be accidentally placed in a wrong position. This rule makes it easier to spot these errors.

1. Don’t use !important;

Using the important tag should be a last resort, as this makes the website harder to maintain and debug. All other possible options should be attempted to apply the style needed, before using !important.

1. Use CSS comments.

Commenting can make code easier to understand, make maintenance much easier, helps make finding bugs faster. You could look at commenting as a form of documentation.

* 1. The terms web libraries and web frameworks are often mistakenly interchange.

Research the terms and clarify the following:

1. What are web libraries and web frameworks?

Web libraries: Sets of code already written, such as frequently used functions, that YOU can call upon when building your own code, and can be used over and over again. This is important in web development to keep code DRY(Don’t repeat yourself).

Web frameworks: A framework can be everything you use in web app development including libraries, in a specific environment for building a website/web application. A framework can provide the starting blocks to your website/web application, in which you build on top of. The main difference between a library and a framework is a framework will call your code, where as your code calls a library.

1. Outline the differences between web frameworks and web libraries.

A framework is a set of pre-written code that provides a structure for developing software applications, where as a library is a collection of pre-written code that can be used to perform specific tasks.

The framework will tell you where to put your code, but will call your code as needed. In the framework it is the code in the framework that calls the code, but in a library, your code calls the code.

A framework defines the overall architecture of the application and provides a common set of conventions for building and maintaining the code. A library is intended to be used as a tool to help developers accomplish specific tasks.

Frameworks can be more rigid and may limit the developer’s ability to customize certain aspects of the application. Libraries are generally more flexible and can be easily replaced or swapped out if necessary.

* 1. Resolve the contingency task situation.

If the tester communicated with me, the developer, that they were ill, I would relay this information to my manager (the project lead) and ask them how they would like to proceed, asking if its possible for 2 of us to test, could we get another tester, or if we would need to reschedule.

I would also ask the tester to communicate their situation with the manager (project lead).

Appendices

References

<https://www.business.qld.gov.au/running-business/digital-business/online-risk-security/legal-obligations>

<https://www.geeksforgeeks.org/explain-different-markup-languages-other-than-html/>

<https://www.shiksha.com/online-courses/articles/framework-vs-library/>