

Student  
Assessment Guide:

ICTWEB441-ICTWEB518 Display data from XML document on a web page using JavaScript

**Copyright 2024**

© Australian College of Business Intelligence

All rights reserved

Version: 24.0

Date Modified: April 2024

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise without the prior written permission of Australian College of Business Intelligence.

**Disclaimer:**

The Australian College of Business Intelligence does not invite reliance upon, nor accept responsibility for, the information it provides. The Australian College of Business Intelligence makes every effort to provide a high-quality service. However, neither the Australian College of Business Intelligence, nor the providers of data, gives any guarantees, undertakings or warranties concerning the accuracy, completeness or up-to-date nature of the information provided. Users should confirm information from another source if it is of sufficient importance for them to do so.

Contents

[**1. Assessment Information 4**](#_heading=h.gjdgxs)

[**2. Assessment Coversheet 8**](#_heading=h.4d34og8)

[**3. Assessment Questions 9**](#_heading=h.2s8eyo1)

[Task A - Perform knowledge test 9](#_heading=h.17dp8vu)

[Task B – Perform requirements analysis on the website and xml document. 13](#_heading=h.3rdcrjn)

[Task C – Design the website and xml documents 14](#_heading=h.lnxbz9)

[Task D – Develop the website and xml documents 15](#_heading=h.35nkun2)

[Task E – Test the website and xml document functionality and security 16](#_heading=h.1ksv4uv)

[Task F – Feedback 17](#_heading=h.44sinio)

[**4. Candidate Self Checklist 18**](#_heading=h.2jxsxqh)

# 1. Assessment Information

## Purpose of assessment

This assessment assesses your knowledge and skills in:

* designing and building an extensible markup document
* producing client-side scripts for creating interactive websites

## What you are required to do

For this assessment, you are required to complete seven tasks:

* Task A – Perform knowledge test
* Task B – Perform requirements analysis on the website and xml document
* Task C – Design the website and xml document
* Task D –Develop the website and xml document
* Task E – Test the website and xml document functionality and security
* Task F – Feedback

All tasks of this assessment require you to use the provided case study information relating to the Bryan’s Café.

## Competencies being assessed

**Elements**

To achieve competency in this unit, the learner must demonstratetheir ability to:

1. Establish and analyse requirements for web documents and xml documents
2. Design and develop web documents and xml documents
3. Test, debug and finalise scripts and xml documents

**Performance Evidence**

Evidence of the ability to:

* design and produce a script according to web document functionality requirements and organisational procedures. With this, the learner must be able to:
  + produce dynamic web page documents, considering accessibility of web page
  + test and debug web document functionality and confirm web document is secure
  + document and seek approval from required personnel.
* produce at least two (extensible markup language) XML documents according to technical requirements. With this, the learner must be able to:
  + test and validate XML document on at least two different browsers and at least two different devices
  + comply with applicable organisational policies, procedures and standards.

**Knowledge Evidence**

The learner must be able to demonstrate knowledge of the following:

* basic principles behind open platform programming
* client-side scripting and its application to dynamic web page design, including:
  + events and event handlers
  + internet operation related to clients
  + internet protocols
  + simple hypertext markup language (HTML)
  + applicable standards
* purpose and differences between server-side and client-side scripting
* standards associated with programming documentation
* script testing methodologies
* cyber security procedures and protocols
* organisational procedures relevant to producing client-side scripts.
* extensible markup language
* unified-modelling language
* software implications for XML programming
* standards impacting XML programming
* design methodologies including software engineering life cycle
* document type definition (DTD)
* XML document components including entities, elements and their attributes
* debugging methods
* document validation and testing procedures
* organisational policies, procedures and standards applicable to building XML documents.

For further information on the competencies of this unit, please refer to:

* <https://training.gov.au/Training/Details/ICTWEB441>
* https://training.gov.au/Training/Details/ICTWEB518

## Important resources for completing this assessment

To complete this assessment, please refer to the following resources provided on Moodle:

* ICTWEB441 Student Guide
* ICTWEB441 self-study guide
* ICTWEB518 learner guide
* ICTWEB518 self-study guide
* ICTWEB441\_518Case study
* ICTWEB441\_518Marking Guide
* Assessment templates

## A note on plagiarism and referencing

Plagiarism is a form of theft where the work, ideas, inventions etc. of other people are presented as your own.   
  
When quoting or paraphrasing from a source such as the Internet, the source must be recognised. If quoting from a source, make sure to acknowledge this by including “quotation marks” around the relevant words/sentences or ideas. Note the source at the point at which it is included within the assessment, such as by using a citation. Then list the full details of the source in a ‘references’ section at the end of the assessment.

All sources used for the assessment should be detailed in a ‘references’ section. It is advisable to never copy another person’s work.

## Instructions for completing this assessment

Answer the questions below using the spaces provided:

* Answer all parts of each question
* Use your own words and give examples wherever possible
* The quality of your answer is more important than how long it is
* Enter your answers in this document

You may use various sources of information to inform your answers, including your resources provided by ACBI, books, and online sources. You must acknowledge and cite your sources.

**Submission via Moodle**

Please refer to the “Instructions for Submitting Your Assessment”found within the unit course page on Moodle.

*NOTE: Please take care to follow all instructions listed. Assessments uploaded with a draft status on Moodle may not be graded.*

# 2. Assessment Coversheet

| **Candidate Name:** | Kadır Boz | | |
| --- | --- | --- | --- |
| **Student ID:** | acbi20240359 | | |
| **Contact Number:** | 0432 477 053 | | |
| **Email:** | k.kadirboz@gmail.com | | |
| **Trainer / Assessor Name:** | Cameron | | |
| **Qualification:** | ICT50220 Diploma of Information Technology – Front end web development | | |
| **Units of Competency:** | ICTWEB441 Produce basic client-side script  ICTWEB518 Build a document using extensible markup language | | |
| **Assessment Tasks:** | ☐ Task A – Perform knowledge test  ☐ Task B – Perform requirements analysis on the website and xml document  ☐ Task C –Design the website and xml document  ☐ Task D – Develop the website and xml document  ☐ Task E – Test the website and xml document  ☐ Task F– Feedback | | |
| **Due Date:** |  | **Date Submitted:** |  |
| **Declaration:** | I have read and understood the following information at the beginning of this assessment guide (please tick): | | |
| ☐ Assessment information  ☐ Submitting assessments  ☐ Plagiarism and referencing | | |
| I declare this assessment is my own work and where the work is of others, I have fully referenced that material. | | |
| **Name (please print):** | Kadir Boz | | |
| **Candidate signature:** | Kadir Boz | | |
| **Date:** | 07/12/2024 | | |

# 

# 3. Assessment Questions

## Task A - Perform knowledge test

| 1.List three basic principles upon which open platform programming is based. |
| --- |
| Open platform programming is built on several fundamental principles that foster innovation, interoperability, and accessibility. Three key principles that form the foundation of open platform programming are:  Adherence to Open Standards: This principle ensures that the platform utilizes publicly available, widely accepted protocols and data formats. By doing so, it promotes transparency and enables seamless integration with other systems and technologies12.  Exposure of Open APIs: Open platforms provide well-documented, publicly accessible Application Programming Interfaces (APIs). These APIs allow developers to interact with the platform's core functionalities, enabling the creation of new applications and services that extend the platform's capabilities34.  Cross-Platform Compatibility: This principle emphasizes the ability of applications developed for the platform to run across different environments with minimal modifications. It promotes flexibility and reduces dependency on specific hardware or software configurations52.  By adhering to these principles, open platforms foster a collaborative ecosystem that encourages innovation, reduces vendor lock-in, and provides greater flexibility for both developers and users.  Sources: 1. <https://blog.ossph.org/programming-principles-every-developer-should-know/>  2. https://apperta.org/assets/Apperta\_Defining\_an\_Open\_Platform\_April.pdf |

| 2.Describe each of the following terms that relate to client-side scripting and its application to dynamic web page design. |
| --- |
| *Write your answer in the table below*   | **Term** | **Description** | | --- | --- | | Events | Events are actions triggered by user interactions, such as clicks or key presses, that can initiate functions within a system.. | | Event handlers | Event handlers are functions that execute in response to specific events, determining what actions to take when those events occur. | | Internet protocols | Internet protocols are standards that dictate how data is transmitted over the internet, with HTTP and HTTPS being key examples for client-server communication. | | Hypertext markup language (HTML) | HTML is the standard language used to create and structure web pages, forming the basic framework that can be enhanced with CSS and JavaScript. | | A standard that applies | The DOM is a programming interface for HTML and XML documents that provides a structured way to access and update the content and style of web pages dynamically. |     *Sources: -* https://www.techtarget.com/searchsoftwarequality/definition/HTTPS *-* [*https://aws.amazon.com/compare/the-difference-between-https-and-http/*](https://aws.amazon.com/compare/the-difference-between-https-and-http/)  *- https://www.w3schools.com/js/js\_events.asp* |

| 3. Explain the purpose and difference of server-side and client-side scripting. |
| --- |
| Server-side and client-side scripting are two fundamental approaches in web development, each with distinct purposes and advantages. Server-side scripting refers to scripts that execute on the web server before any content is delivered to the user's browser. The primary role of server-side scripting is to manage backend logic, interact with databases, and perform sensitive operations that require secure handling.On the other hand, client-side scripting runs directly in the user's browser, focusing on enhancing interactivity and improving the overall user experience. This type of scripting is responsible for tasks such as validating forms and updating content dynamically without necessitating a full page reload.The key difference between the two lies in their execution environment: server-side scripts are concealed from users and have access to server resources, while client-side scripts are visible to users and operate within the constraints of the browser. Essentially, server-side scripting addresses backend processes, whereas client-side scripting enhances frontend functionality, allowing for a more engaging user interface.  Source: https://www.geeksforgeeks.org/difference-between-server-side-scripting-and-client-side-scripting/ |

| **4.Describe two standards relevant to programming languages in general.**  *Answer in 40-80 words.* |
| --- |
| ISO/IEC 14882 (C++ Standard) defines the C++ programming language, ensuring consistency across implementations. It specifies syntax, semantics, and standard libraries for C++. The standard has evolved through versions like C++98, C++11, C++14, C++17, and C++20, introducing new features and improvements to enhance the language's capabilities and performance.  ANSI/ISO/IEC 9075 (SQL Standard) defines the Structured Query Language for managing relational databases. It specifies syntax and semantics for database operations, ensuring consistency across different database management systems. The standard has evolved through multiple versions, introducing new features and capabilities for data manipulation, querying, and database management  Source: https://www.open-std.org/jtc1/sc22/wg14/www/docs/n3220.pdf |

| **5. List two methods that can be used to test scripts within a website.**  *Answer in 40-80 words.* |
| --- |
| Two methods for testing scripts within a website are:   1. Console Testing: Developers can use the browser's console to execute JavaScript code directly, allowing them to test individual functions, debug issues, and verify script behavior in real-time. 2. Unit Testing: This involves writing automated tests for specific functions or components of the website's scripts. Tools like Jest or Mocha can be used to create and run these tests, ensuring individual parts of the code work as expected.   These methods help developers identify and fix issues in website scripts, improving overall functionality and user experience.  Source: https://developer.mozilla.org/en-US/docs/Learn/Tools\_and\_testing/Cross\_browser\_testing/JavaScript |

| **6. Describe the purpose of extensible markup language.**  *Answer in 40-80 words.* |
| --- |
| Extensible Markup Language (XML) serves as a versatile tool for structuring, storing, and transmitting data. Its primary purpose is to provide a format that is easily understood by both humans and computers. XML allows for the creation of custom tags, enabling developers to organize information hierarchically according to specific needs. This flexibility makes XML ideal for diverse applications, including web services, configuration files, and document storage. Its straightforward nature and adaptability have established XML as a widely adopted standard for data exchange across different systems and platforms.  Source: https://www.w3schools.com/xml/xml\_whatis.asp |

| **7. Describe unified modelling language (UML).**  *Answer in 40-80 words.* |
| --- |
| Unified Modeling Language (UML) is a versatile visual language used in software development to map out system designs. It offers a standardized set of diagrams that help developers and stakeholders visualize, plan, and communicate complex software architectures. UML includes various diagram types, such as class diagrams for showing relationships between objects, sequence diagrams for illustrating interactions over time, and use case diagrams for depicting system functionality from a user's perspective. This visual approach makes it easier to understand and discuss system designs, bridging the gap between technical and non-technical team members.  Source: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/overview-of-the-14-uml-diagram-types/ |

| **8. Give an example of a software that can be used for creating XML documents.**  *Answer in 40-80 words.* |
| --- |
| **One popular software for creating XML documents is XML Notepad, a free and user-friendly tool developed by Microsoft. This lightweight editor offers a simple interface with a tree-view structure, making it easy to navigate and edit XML files. XML Notepad provides features like syntax highlighting, validation, and IntelliSense, which help users create well-formed XML documents efficiently. It's particularly useful for beginners and those who need a straightforward tool for occasional XML editing without the complexity of more advanced editors.** |

| **9. Complete the following table, listing two standards that impact XML programming and how.**  *Answer in 40-80 words.* |
| --- |
| *Write your answer here*   | Standards that impact on XML programming | How the standard affects XML programming | | --- | --- | | XML Publishing:  <https://www.w3.org/standards/xml/publishing> | This standard provides essential guidelines for how to transform and present XML content effectively. It includes tools like XSLT and XSL-FO, which help developers turn XML data into various formats for web pages, printed materials, and more. This versatility makes it easier to use XML documents in different contexts, enhancing their usability.. | | Security Standards:  <https://www.w3.org/standards/xml/security> | These standards tackle important security issues related to XML, focusing on aspects like XML Encryption and XML Signature. They offer methods for securely transmitting and storing XML data, allowing developers to encrypt sensitive information and ensure that XML documents are authentic and intact. This is vital for protecting data privacy in XML-based applications and web services. | |

| **10. Describe the software development life cycle and its relevance to design methodologies.**  *Answer in 40-80 words.* |
| --- |
| The Software Development Life Cycle, or SDLC for short, is like a roadmap for creating software. It's a step-by-step approach that guides developers from the initial idea all the way to the finished product and beyond. Think of it as a recipe for cooking up great software.This cycle typically includes stages like figuring out what the software needs to do, sketching out its design, actually building it, making sure it works properly, getting it out to users, and then keeping it running smoothly.What's cool about SDLC is how it ties into different design approaches. It's like the big picture that helps teams organize their design process within the larger scope of development. By following this cycle, developers can make sure they're thinking about design at every stage, from the first brainstorming session to the final tweaks and updates after launch.  Source: https://slash.co/articles/complete-explanation-of-the-software-development-life-cycle-sdlc-and-its-5-stages |

| **11. Explain the term document type definition (DTD).**  *Answer in 40-80 words.* |
| --- |
| Think of a Document Type Definition (DTD) as a rulebook for XML documents. It's like a blueprint that tells you what pieces you can use and how to put them together when you're building an XML document.DTDs spell out what elements and attributes are allowed, and how they should be arranged - kind of like explaining the rules of a game before you start playing. This helps make sure everyone's on the same page when sharing XML data.You can either include these rules directly in your XML file or keep them in a separate file. Either way, DTDs act like a quality control checklist, making sure your XML documents are structured correctly and consistently. It's a handy tool for keeping things organized and making sure different systems can understand each other's XML data.  Source: <https://www>.geeksforgeeks.org/what-is-dtd-in-xml/ |

| **12. Describe each of the following XML document components.**  *Answer in 40-80 words.* |
| --- |
| *Write your answer in the table below*   | Entities | Think of entities as shortcuts or placeholders in XML. They're like little code words that stand in for longer text or special characters. Some are built-in, like "&lt;" for "<", while others you can make up yourself. They're handy for keeping things consistent and making updates easier. You use them by putting an "&" before the entity name and a ";" after it. | | --- | --- | | Elements | Elements are the building blocks of XML, like LEGO pieces for your document. They usually have an opening tag, some content in the middle, and a closing tag - imagine a sandwich with the tags as bread and the content as filling. Elements can nest inside each other, creating a family tree of data. They're the main way to structure information in XML. | | Attributes | Attributes are like name tags for elements, giving extra details about them. They live inside the opening tag of an element and come in pairs - a name and a value. It's a quick way to add simple extra info without creating a whole new element. Think of them as sticky notes attached to your elements, providing bonus information at a glance. |   *Source:  1. https://www.w3schools.com/xml/xml\_tree.asp*  *2. https://www.w3schools.com/xml/xml\_attributes.asp* |

| **13. List two debugging methods that can be used for XML.**  *Answer in 40-80 words.* |
| --- |
| *Write your answer here*   | Method 1: | Think of XML Debugger tools as a magnifying glass for your XML code. They let you peek inside your XML processing, like watching a movie in slow motion. You can pause at any point, check what's happening, and even set up flags (we call them breakpoints) to stop at interesting parts. These tools often show your XML structure like a family tree, making it easier to spot where things might be going wrong. | | --- | --- | | Method 2: | XML Validation is like a spell-checker for your XML. It compares your XML document against a set of rules (called schemas or DTDs) to make sure everything's in order. It catches typos, missing pieces, and parts that don't fit together right. It's like having a proofreader who knows exactly how your XML should look, helping you fix mistakes before they cause bigger problems. |   *Sources:* https://www.ibm.com/docs/en/clearcase/9.0.1 |

| 14. Describe the process for XML document validation. |
| --- |
| Imagine you're checking a big, complex puzzle to make sure all the pieces fit correctly. That's kind of what XML validation is like. Here's how it works  First, we do a quick once-over to make sure the puzzle pieces are all there and facing the right way up. In XML terms, this is called a well-formedness check. We're just making sure the basic rules are followed, like closing tags properly.  Next, we grab the puzzle box lid - that's like our schema or DTD. It shows us what the finished puzzle should look like.  Now comes the fun part. We start comparing our puzzle to the picture on the box. Are all the pieces in the right place? That's our structure validation. We're checking if everything is where it should be.  As we go, we also make sure each piece is the right shape and color. In XML, this is like checking data types. Is a number where a number should be? Is a date formatted correctly?  Sometimes, the puzzle box might have special instructions, like "the blue sky pieces must connect first." These are like additional constraints in XML. We check for those too.  If we find any problems along the way - maybe a piece is missing or in the wrong spot - we make a note of it. That's our error reporting.  By the end of this process, we know if our XML "puzzle" is put together correctly or if it needs some fixing up.  Sources: https://www.w3schools.com/xml/xml\_validator.asp |

| 15. Describe the testing process that can be used for XML documents. |
| --- |
| Think of testing XML documents like giving a thorough health check-up to a complex machine. Here's how we go about it:  First, we do a quick once-over, making sure all the parts are there and in the right place. This is our well-formedness check. It's like making sure all the gears in our machine are present and not jammed.  Next, we compare our XML to its blueprint - that's the schema or DTD. It's like checking if our machine matches the manufacturer's specifications. Does everything line up as it should?  Then we dive into the details. We're looking at the actual data inside, making sure it makes sense and fits the purpose. It's like checking if our machine is filled with the right oil and coolant.  Now, we turn the machine on and see how it runs. This is our functionality testing. Does our XML work properly when we use it in our systems? Can we easily get the information we need from it?  If we're dealing with a big, complex XML or need to process lots of them quickly, we'll do a speed test. This is like seeing how our machine performs under pressure.  Finally, if our XML is going to be used on different devices or browsers, we give it a spin on each one. We want to make sure it works smoothly no matter where it's used, just like we'd want our machine to run well in different environments.  This process helps us catch any issues early and ensures our XML is in top shape for whatever job it needs to do  Source: https://www.w3.org/standards/ |

## Task B – Perform requirements analysis on the website and xml document.

**Case Study**

Please read the case study provided for this unit to complete tasks B to G. In this assessment, you will play the role of web developer, while your trainer will be your systems analyst.

You need to perform requirements analysis on Bryan’s Café website and XML documents. To achieve this, you need to:

* Analyse the case study provided.
* Based on your analysis, create a requirements analysis report using the template with filename: taskBRequirementsAnalysis.docx. This template will be provided in the learning management system. Follow all instructions given in this template and provide the required details on all sections.
  + Note that a meeting with the systems analyst will take place. The schedule will be provided to you.

## Task C–Design the website and xml documents

**Website Design**

In this activity, you are to design the website based on the client’s requirements using wireframe or mock-up.

**The design must be submitted in pdf format with the filename: taskCWebDesign.pdf.**

**XML Tree Structure**

Provide a diagram of the tree structure for each XML document. This can be hand drawn here or developed on a computer in which case you should attach a screenshot.

The **tree structure must be submitted in pdf format** with the filename: taskCTreeStructure.pdf.

## Task D – Develop the website and xml documents

Based on the case study, create a website project and name it as: bryanCafe. The project should contain the following HTML documents having the following filenames:

| **HTML Document** | **Filename** | **Page Title** |
| --- | --- | --- |
| About | index.html | About |
| Menu | Menu.html | Menu |
| Contact | Contact.html | Contact |

Apply the layout, colour and font styling to each of the web pages based on the design and case study.

Once you have completed this task:

* Submit the website project as evidence that you have met the requirements for this task.
* Host the website

## Task E–Test the website and xml document functionality and security

You need to test the functionality and security of Bryan’s Café website and XML documents. To achieve this, you need to:

* Complete task D.
* **Create the test document using the template with filename: taskESecurityAndTesting.docx**. This template will be provided in the learning management system. Follow all instructions given in this template and provide the required details on all sections.

## Task F – Feedback

You need to send an email to your systems analyst asking for approval of your website. The email should include:

* link to your website
* Link to both of your xml and dtd documents

Based on the XML and DTD documents that you’ve provided, the systems analyst will:

* Look for errors on your XML and DTD documents and request you to fix the errors
* If there are no errors, you will be provided with an example for you to fix.

Take screenshots of the work you undertook to fix errors.

# 4. Candidate Self Checklist

**Candidate Self Checklist for Tasks A - F**

| **Candidate name:** |  | |
| --- | --- | --- |
| **Unit of Competency:** | ICTWEB441 Produce basic client-side script  ICTWEB518 Build a document using extensible markup language | |
| **Instructions:** *Place a tick ‘✓ ’ in the Yes (“Y”) column for each question you have completed all parts for.* | | |
| **Task A –Perform knowledge test.** | | |
| **Did you:** | | **Y**  **✓** |
| *1. List three basic principles upon which open platform programming is based.* | |  |
| *2. Describe the required terms that relate to client-side scripting and its application to dynamic web page design.* | |  |
| *3. Explain the purpose and difference of server-side and client-side scripting.* | |  |
| *4. Describe two standards relevant to programming languages in general.* | |  |
| *5. List two methods that can be used to test scripts within a website.* | |  |
| *6. Describe the purpose of extensible markup language.* | |  |
| *7. Describe unified modelling language (UML).* | |  |
| *8. Give an example of a software that can be used for creating XML documents.* | |  |
| *9. List two standards that impact XML programming and how.* | |  |
| *10. Describe the software development life cycle and its relevance to design methodologies.* | |  |
| *11. Explain the term document type definition (DTD).* | |  |
| *12. Describe the required XML document components.* | |  |
| *13. List two debugging methods that can be used for XML.* | |  |
| *14. Describe the process for XML document validation.* | |  |
| *15. Describe the testing process that can be used for XML documents.* | |  |

| **Task B – Perform requirements analysis on the website and xml document** | |
| --- | --- |
| **Did you perform the following on your requirements analysis report?** | **Y**  **✓** |
| *Outline the client requirements for the website based on your initial review of the documentation.* |  |
| *Outline the procedures you need to follow to produce websites that includes include language in which the website is to be developed and cyber security procedures and protocols to be followed in relation to website development.* |  |
| *Write down your questions and your client’s response to address the questions?* |  |
| *Outline the client requirements for the website based on your meeting with the client.* |  |
| *Identify:*   * *One applicable legislation and its relevance to XML document development* * *Two applicable standards and its relevance to XML document development* |  |
| *Outline the procedures that you will follow to develop your XML documents* |  |
| *Describe the two XML documents you are going to design and develop, including:*   * *purpose* * *expectations for the document* * *required functionality.* |  |
| *Document the design methodology that you will use for your XML documents.*  *Explain how the methodology will incorporate iterative development.* |  |
| *For each XML document, define the:*   * *entities* * *elements* * *attributes (if necessary)* |  |

| **Task C – Design the website and xml document** | |
| --- | --- |
| **Did you perform the following?** | **Y**  **✓** |
| *Design the website based on the client’s requirements using wireframe or mock-up.* |  |
| *Provide a diagram of the tree structure for each XML document* |  |

| **Task D – Develop the website and xml document** | |
| --- | --- |
| **Did you:** | **Y**  **✓** |
| *Create the website and xml documents based on the customer requirements?* |  |
| *Host the website?* |  |

| **Task E – Test the website and xml document functionality and security** | |
| --- | --- |
| **Did you:** | **Y**  **✓** |
| *Test the website in order to:*   * *Determine that the website performs all the required functionality* * *Check that the website is secure and bug free* |  |
| *Describe the testing process you followed to ensure that:*   * *The website is functional* * *The website is functional after the change you made* * *The website complies with cyber security procedures. Describe how you follow the procedures.* |  |
| *Test both of your XML documents offline to check if they are working on two different browsers.* |  |
| *Describe the testing process you undertook and paste the screenshots below.* |  |
| *Test that data from xml documents appear in menu and contact page on laptop and mobile phone.* |  |
| *Test both of your XML documents online to check if they are working* |  |

| **Task F – Feedback** | |
| --- | --- |
| **Did you:** | **Y**  **✓** |
| *Send an email to your systems analyst asking for approval of your website that includes:*   * *link to your website* * *Link to both of your xml and dtd documents* |  |
| *Fix the XML and DTD errors* |  |