Web Developer Apprenticeship Portfolio Evidence

**Project Name**

|  |  |
| --- | --- |
| Apprentice Full Name |  |
| Employer |  |
| Signature |  |
| Date |  |

Contents

[**Introduction - How to use this document**](#_3tgn5di68p7o) **3**

[Think about Context](#_carnwgaadxhk) 3

[Think about Narrative](#_qcqr6hca1nwk) 3

[**Context**](#_afarjm2u9id) **4**

[**Planning**](#_ju4j7rqj5rtt) **4**

[**Design**](#_qtx7ckf033my) **4**

[**Development**](#_hrky1o5z40dd) **4**

[**Testing**](#_piwwkfb2ev7y) **4**

[**Deployment**](#_ejewdqk97vch) **4**

[**Maintenance**](#_idcx323t2uv8) **5**

[**Reflection**](#_42sg66cp8xr9) **5**

[**Appendix A - Standard Checklist**](#_hu6kqbdg2h3d) **5**

[COMPETENCIES (Skills)](#_gqrikwkt5hwe) 5

[BEHAVIOURS (Skills)](#_vbuyvhqzxihh) 6

[**Appendix B, C, D etc.**](#_xpcg5oz1ukj2) **6**

# 

# Introduction - How to use this document

This section has information and suggestions on how to provide your evidence. The section headers have suggested subheadings but they are just examples and you should not feel restricted - some may not be necessary for your situation and you should ad whatever additional detail you need.

## Think about Context

The assessor won’t know anything about you or your organisation, write a short description of your organisation and your role. You can repeat this for all statements.

e.g. “...I work for xxx , a full service marketing agency who design and produce websites as part of their service offering. The company has 30 employees, 3 C-level, 2 in operations, 6 account managers, 5 in design, 20 in marketing and 12 in the development team. My role is front-end developer, within the dev team we have 6 front-end specialists. We are responsible for….”

Add some information about the activity/task/project, what is the task/activity/project, who gave it to you, who is involved, who are/is the ‘customer’, what is the time scale etc.

e.g. “...I was given the task of prototyping an app for team leaders and managers to enter employees overtime...”

## Think about Narrative

Write a description of how you carried out the task, a narrative form works best. What tools did you use, how did you use them, who did you work with, who did you communicate with, how did you communicate etc. what knowledge did you use. The statement should include justification of what you have done and proof of what you have done.

Remember that the assessor won’t know anything about what resources you have available, what process you have to follow etc. e.g. if you have to use a certain tool, then make sure you state that it is the only tool available to you. You can enhance this by discussing the pros and cons of the tool and alternatives that could be used.

If you are dealing with confidential information, then remember to state the process that you have to follow and show some evidence of following them. The same for H&amp;S if appropriate.

Include screenshots, photos etc. in the statement, reference attached additional evidence such as recordings, documents etc. Do not link to emails or documents in statement.

Include feedback from others, particularly your line manager, this not only provides proof that you have carried out the activity but also gives an opportunity for examples of behaviour. Make sure that you include contact details for witnesses.

Do be careful of SPAG (Spelling, punctuation and Grammar) make sure you proofread before submitting. For example, when using abbreviations/acronyms make sure they are expanded the first time they are used, such as TLA (Three Letter Acronym).

Include references to appropriate Unit/Criteria, e.g. “...I sent an email to my manager giving him the results. (C4)...”

For an example of structure consider that there is a beginning, middle and end. The beginning is how you have planned the activity, the middle is how you carried out the activity and the end is the results. The end can also include some reflection e.g. what lessons did you learn (reflection), what would you do differently next time?

Think in terms of - **WHAT, HOW, WHY, WHO…**

## Remove these pages

Once this evidence doc is complete you should remove this introduction section and update the Table of Contents.

# 

# Context

* Outline
* Objectives
* Background

# Planning

* Timescales & Cost
* Software Methodology Approach
* Key Colleagues Involved
* User Requirements
* Functional Requirements
* Non-Functional Requirements

# Design

* Design Guidelines
* Assumptions
* Prototyping
* Wireframes

# Development

* Version Control
* Problems and Resolutions
* Documentation

# Testing

* Test plan
* Results and Documentation

# Deployment

* Methods
* Problems and Resolutions

# Maintenance

* Methods
* Problems and Resolutions

# Reflection

* What went well
* What could have gone better
* What I would do differently

# Appendix A - Standards Checklist

## COMPETENCIES (Skills)

|  |  |  |
| --- | --- | --- |
| C1 |  | Logic: writes good quality code (logic) with sound syntax in at least one language. |
| C2 |  | User interface: can develop effective user interfaces for at least one channel. |
| C3 |  | Data: can effectively link code to the database/data sets. |
| C4 |  | Test: can test code and analyse results to correct errors found using either V-model manual testing and/or using unit testing. |
| C5 |  | Problem solving: can apply structured techniques to problem solving, can debug code and can understand the structure of programmes in order to identify and resolve issues. |
| C6 |  | Design: can create simple data models and software designs to effectively communicate understanding of the program, following best practices and standards. |
| C7 |  | Analysis: can understand and create basic analysis artefacts, such as user cases and/or user stories. |
| C8 |  | Deployment: can understand and utilise skills to build, manage and deploy code into enterprise environments. |
| C9 |  | Development lifecycle: can operate at all stages of the software development lifecycle, with increasing breadth and depth over time with initial focus on build and test. |
| C10 |  | Can apply good practice approaches according to the relevant paradigm (for example object oriented, event driven or procedural). |
| C11 |  | Can interpret and follow:  − software designs and functional/technical specifications  − company defined ‘coding standards’ or industry good practice for coding  − testing frameworks and methodologies  − company, team or client approaches to continuous integration, version and source control |
| C12 |  | Can respond to the business environment and business issues related to software development. |
| C13 |  | Can operate effectively in their own business’, their customers’ and the industry's environments. |
| C14 |  | Can apply the maths required to be a software developer (e.g. algorithms, logic and data structures). |

Indicate in Column 2 where competencies have been covered (**X**).

## 

## BEHAVIOURS (Skills)

|  |  |  |
| --- | --- | --- |
| B1 |  | Logical and creative thinking skills. |
| B2 |  | Analytical and problem-solving skills. |
| B3 |  | Ability to work independently and to take responsibility. |
| B4 |  | Can use own initiative. |
| B5 |  | A thorough and organised approach. |
| B6 |  | Ability to work with a range of internal and external people. |
| B7 |  | Ability to communicate effectively in a variety of situations. |
| B8 |  | Maintain productive, professional and secure working environment. |

Indicate in Column 2 where competencies have been covered (**X**).

## Knowledge

|  |  |  |
| --- | --- | --- |
| **K1** |  | Understands and operates at all stages of the software development lifecycle. |
| **K2** |  | Understands the similarities and differences (taking into account positives and negatives of both approaches) between agile and waterfall software development methodologies |
| **K3** |  | Understands how teams work effectively to produce software and contributes appropriately. |
| **K4** |  | Understands and applies software design approaches and patterns and can interpret and implement a given design, compliant with security and maintainability requirements. |
| **K5** |  | Understands and responds to the business environment and business issues related to software development. |
| **K6** |  | Understands and applies the maths required to be a software developer (eg algorithms, logic and data structures). |

# 

# Appendix B, C, D etc.

Example documentation