Making IT
good for society

BCS Digital Industries Apprenticeship

Software Development Technician
Project A - Media Organiser Overview

Version 1.0

March 2018

Change History

Any changes made to the project shall be clearly documented with a change history log. This shall include the latest version number, date of the amendment and changes made. The purpose is to identify quickly what changes have been made.

Version Number and Date	Changes Made
V1.0 16/03/18	Document created.

Project Overview and Objectives

You work for Whizzy Software, a software house specialising in serving the needs of clients in the media industry, such as TV and radio companies, music streaming services, etc. One of their clients needs a stand-alone component that supports the organisation of media files on a device such as a desktop computer, laptop, tablet or smartphone. Your manager would like you to design, build, and test an initial version of this component – the "Media Organiser".

An apprentice will need to:

1. **Review all the key information and create a design for the media organiser;**
2. **Construct the media organiser in accordance with the design;**
3. **Test that the media organiser meets its requirements;**
4. **Document what you built.**

Project Outputs and Deliverables

Once completed, to demonstrate completion of the tasks you will be asked to provide a series of outputs that should be submitted together with the synoptic project declaration.

Deliverable	Output	Evidence
Design	Create documentation to describe what the media organiser will do and how it will work. This is likely to include: • Any assumptions made about the requirements or changes made to the requirements; • Sketches of the user interface; • Brief explanations describing what each element of the user interface does; • A specification of the format of data saved and loaded by the component.	Word or PDF documents or similar
Construction	Write a program that implements the media organiser. • Your program should be logically structured. • It should follow good coding practices.	Files containing program code
Test	Create and execute a set of tests that demonstrate that the program meets its requirements. • The tests may be manual or automated. • The tests may be written before, after, or at the same time as the program code. • For each test you should document its expected outcome and the actual result.	Any suitable format e.g. textual documents, spreadsheets, program code.
Document	Document the results of your work. • Discuss any limitations of your design and/or implementation. • Propose future improvements. • Create a user guide.	Word or PDF document or similar. A video might be suitable for a user guide.

Project Information and Equipment

To complete this project, you will need to review all the information in the bullet list below. This can be found in the Appendix and will enable you to deliver the key outputs and deliverables for this project as detailed in the table above.

- Background information.
- Conceptual model.
- Use cases.
- Apprentice declaration.

In addition, you will be provided with access to a virtual platform or alternatively if a virtual platform is not available, your training provider and or employer will provide you with all resources required to complete your project including:

- computer equipment with access to the Internet;
- an appropriate software development environment;
- suitable document preparation software.

Apprenticeship Competencies Covered

Competency Standard
Logic: writes simple code for discrete software components following an appropriate logical approach to agreed standards (whether for web, mobile or desktop applications)
Data: makes simple connections between code and defined data sources as specified.
Test: functionally tests that the deliverables for that component have been met or not.
Analysis: follows basic analysis models such as use cases and process maps.
Quality: follows organisational and industry good coding practices (including those for naming, commenting etc.).
Communication: clearly articulates the role and function of software components to a variety of stakeholders (including end users, supervisors etc.).
User Interface: develops user interfaces as appropriate to the organisations development standards and the type of component being developed.