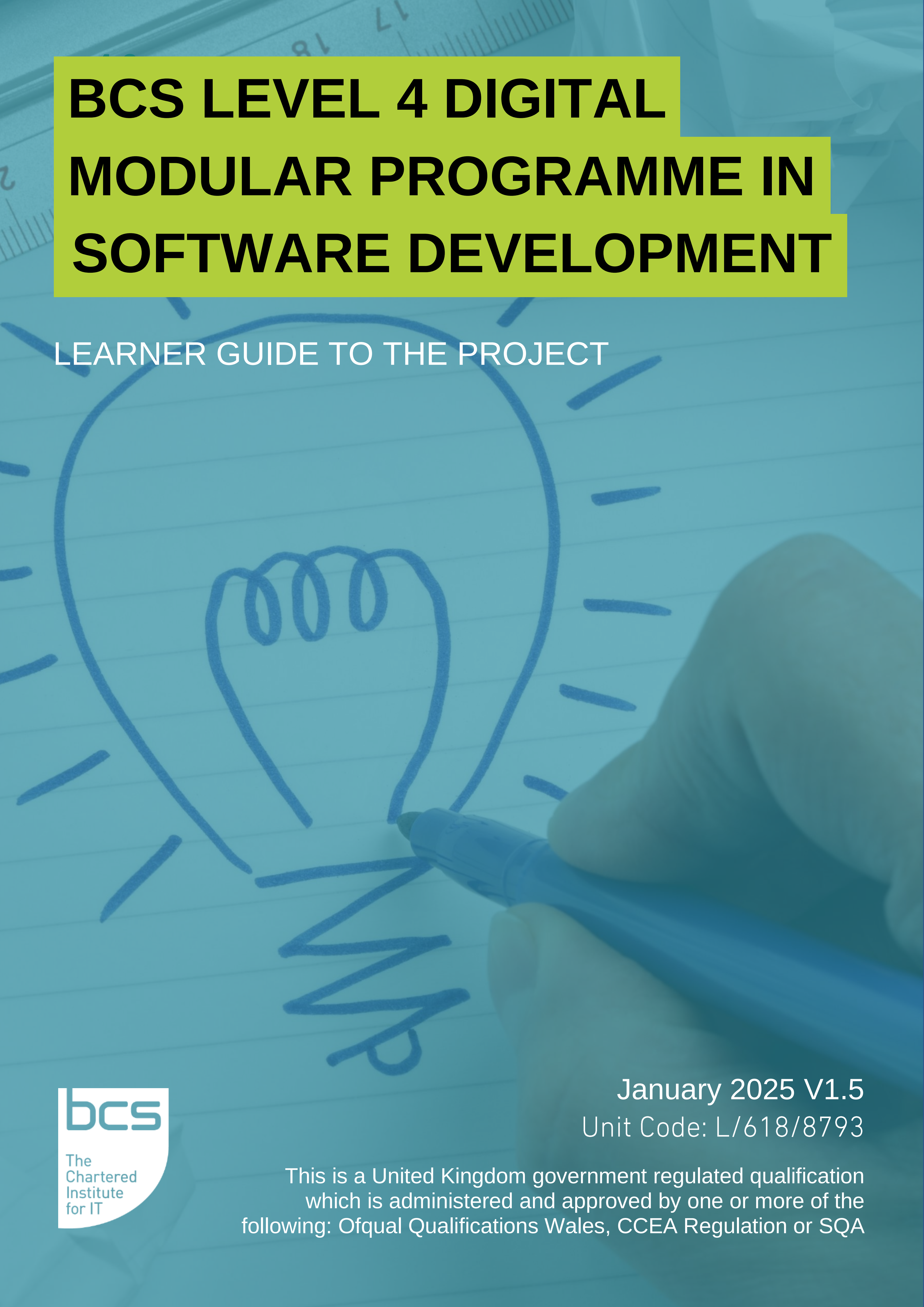

BCS LEVEL 4 DIGITAL MODULAR PROGRAMME IN SOFTWARE DEVELOPMENT

LEARNER GUIDE TO THE PROJECT

January 2025 V1.5

Unit Code: L/618/8793

This is a United Kingdom government regulated qualification which is administered and approved by one or more of the following: Ofqual Qualifications Wales, CCEA Regulation or SQA

Are you ready to showcase your skills?

Well done for all of your hard work so far on the Digital Modular Programme (DMP). It’s now time to begin planning and preparing your synoptic project. This is your opportunity to apply everything you have learned so far, and we want to make sure that you are as prepared as possible.

Within this document, you will find information about the requirements and format for this part of the assessment for the BCS Level 4 Diploma Digital Modular Programme in Software Development.

NB: This synoptic project is part of the mandatory assessment strategy for all qualifications in the DMP suite of qualifications. This project **will not be required** by learners accessing any element of the DMP as part of an Apprenticeship.

Overview of the project

Working in a project-based environment, software developers work through the software development lifecycle to develop a software solution which meets the customer and end-user requirements.

This project will enable you to demonstrate your ability to design, build and test a software solution which fulfils the given requirements. It requires you to draw upon experience gained during your day-to-day work or training to design a project which addresses a software development task that is relevant to your role, or to a given scenario if you are not currently working in a software development role.

Top tip: Start with the end in mind! We encourage you to familiarise yourself with the project early on in your DMP journey and to think about areas or topics that you might like to explore further when undertaking the project.

What should I focus on within my project?

The project should address a specific problem, recurring issue or idea/opportunity. There are two options available for how you may undertake this project, depending on whether you are currently employed in a software development role or not:

1. **A real-world business solution.** Your choice of project will focus on a real work situation, relevant to your current job role. This may include either live client work (which can be anonymised), or work that forms part of a larger-scale or wider focused business activity. The project should set out to solve a specific business problem, recurring issue or idea/opportunity (e.g. to improve the customer experience, or to support a particular operational issue) where you have identified an opportunity to develop a software application that supports a particular aspect of the business. Your project should be based on a set of business requirements that you have gathered.

   You should ensure that you are able to clearly define the problem you wish to solve, who this solution has been designed to support, the type of software and programming languages you will use to develop your solution, and the process through which you will plan, design, build, test and deploy your software application. It is important that you get exposure to all stages of the software development lifecycle so that you can evidence the full range of assessment criteria.

2. **A solution to a given scenario**. You can undertake your project using a given scenario and set of requirements provided by BCS. You will be required to gather further requirements from other stakeholders, such as your tutor, to incorporate into the design of your solution.

   You should ensure that you are able to clearly articulate the problem you are being tasked to solve, understanding who this solution has been designed to support, the type of software and programming languages you will use to develop your solution, and the process through which you will plan, design, build, test and deploy your software application.

The project will follow the same structure whether you choose to focus on a real-world solution or a given scenario. Whichever option you choose, your project should:

- Clearly state the business problem and requirements for software development
- Analyse the requirements for the intended stakeholders to inform on the design of the solution
- Allow others to understand how the solution will work by providing a clear design concept
- Demonstrate your approach to software development through the creation of a working application and evidence of your code
- Demonstrate your approach to testing the solution to provide assurance of its suitability against the requirements

How should I structure my project and what should it include?

The project must represent a substantial piece of work; as such, the suggested time requirement for this part of the assessment is 30 hours. These hours are included in the Total Qualification Time (TQT) of 427 hours. You do not have to be supervised when you complete all the project work but you may wish to have your tutor available for any questions.

We highly recommend that a mentor/tutor supports you throughout your project. When you meet with them you will complete a log of your meetings to track the progress and feedback of your project where each party signs and dates to allow the log to be submitted alongside your project. This will enable you to capture any feedback given to you by your mentor/tutor on your individual performance, how you have worked with others, and provide opportunity for you to record how you have overcome any challenges during each stage of the project.

To ensure you meet the requirements of this assessment, the files you submit for your project should reference the Software Development Lifecycle throughout, as shown on page 4. The guidance provided on pages 6-11 clearly identifies the scope of criteria you need to include within your work for each of the sections of your project.

You should set out your project as follows, with each section containing the relevant content, as a minimum:

Project section	Content to include, as a minimum
1. Identify the problem to be solved	An introduction that includes: • Researching the problem to be solved • The scope of the project (including key performance indicators) • A project plan • Desired project outcomes • Findings
2. Requirements analysis	• A clear requirements statement based on stakeholder discussions and a set of user stories • Analysis
3. Design	• A technical specification and design concept for your solution • Consideration of legislation, regulation, industry and organisational policies, procedures and requirements
4. Code development	• An instance of the built solution, including the interface and backend • Samples of your code with explanatory notes
5. Testing	• A review or report capturing evidence of how you have tested the solution
6. Conclusion and further improvements	• How the outcomes were achieved • Recommendations and conclusions • A reflective account to review the outputs and own performance

A word limit of **3500 words** applies to your project, with a 10% tolerance either way. Any additional information, research data, test logs, samples of work or supporting documentation can be included as an annex, which will not contribute to the overall word count. If the project exceeds the tolerance allowed, then the project will be returned unmarked and a restructure before the second submission would be requested.

The decision on how to use the word limit across the project is entirely yours, but an illustrative example of the breakdown of words by section is provided below to help you, should you wish to use it. Please note that these are only example figures.

Section	Percentage of total word count
1. Identify the problem to be solved	15%
2. Requirements analysis	10%
3. Design	20%
4. Code development	30%
5. Testing	15%
6. Conclusion and further improvements	10%

Top tip: These elements are not equally weighted – this means that you would not be expected to spend the same amount of time on or produce as much for each of the six sections. The percentage of total word counts given above are not reflective of the amount of time that might be spent on each area of the project.

The project must map, in an appendix:

- How it evidences the relevant criteria for this assessment method (as described on pages 6-11)
- Where you have provided justifications for any decisions made during the project

How will I submit my work?

Your project must be submitted electronically and contain notes and guidance for the assessor so that they can easily identify where you have met the learning objectives and assessment criteria in your work. The final submission must be in a format that is straightforward to access, so that your work can be assessed against the stated criteria – these might include PowerPoint, Word, PDF or Excel.

What will I need to demonstrate to pass this assessment?

The following table outlines the tasks you will be required to undertake for each stage of your project, as well as the pass and distinction criteria used for this assessment, with reference to the knowledge, skills and behaviours (KSBs) you will be able to demonstrate by undertaking this project. These KSBs are listed in full in the Occupational Standard section of this document. The syllabus criteria has been provided to show you how the learning undertaken through completion of the Digital Core module and your occupationally-focused module will support you to meet these criteria.

1. Identify the problem to be solved

You will describe the problem to be solved, the solution you intend to develop, and the approach you will take to develop it. You will be required to:

a. Provide an overview of the project you will undertake;
   - i. identifying the audience, users and outputs required. (B8)
   - ii. identifying the scope for the work to be completed and controls to be used which may include security of code, testing and versioning. (K1, S11, S17)
   - iii. identifying the stakeholders with whom you will be working e.g. mentor, colleagues, tutor. (B8)

b. Develop a plan for each stage of the process you are going to undertake following the software development lifecycle. This should also include the resources you will need to complete the task. (B2, B6, B8)

Pass criteria You can demonstrate that you are able to:	**Distinction criteria** You can demonstrate that you are able to:	**Reference to learning** The learning towards this criteria can be supported within the following syllabus areas:
Identify and describe the problem to be solved and explain how software development can present a solution.	Evaluate the roles of others in the project lifecycle and the dependencies that exist to deliver the required solution.	BCS Level 4 Module in Digital Core: 1.5, 2.2

Illustrate plans to undertake each of the stages of the software development lifecycle. Show an understanding of your role within the project environment.	Analyse the cost and benefits of the solution, proposing the return on investment.	BCS Level 4 Module in Software Developer: 1.1, 1.2, 1.3, 2.1, 2.2

2. Requirements analysis

You will outline the requirements of the business and the intended users, illustrating how your solution will meet these requirements. You will be required to:

a. Provide evidence of how you have gathered, analysed and interpreted additional stakeholder requirements to be used within the design of the software application.
b. Develop a set of user stories to articulate the requirements of the intended user and create simple explanatory designs. (S8, S9, B7)

You should include copies of any user surveys or other information used in your requirements gathering within your annex.

Pass criteria You can demonstrate that you are able to:	**Distinction criteria** You can demonstrate that you are able to:	**Reference to learning** The learning towards this criteria can be supported within the following syllabus areas:
Demonstrate an ability to gather, analyse and interpret stakeholder requirements to propose a software solution to address a specific business need. Apply suitable requirements elicitation and analysis approaches to create user stories to articulate the requirements of the intended user.	Categorise the gathered requirements based on priority level and with clear justification.	BCS Level 4 Module in Software Developer: 1.2, 2.1, 2.2, 2.3 BCS Level 4 Module in Digital Core: 7.1

3. Design

You will design your solution with consideration to any particular industry and organisational standards, as well as any technical, legal or regulatory requirements. You will demonstrate the design approach you have taken, using materials to provide visual evidence of your design, ensuring the design concept can be easily understood by both technical and non-technical audiences. You will be required to:

a. Create a technical specification for your solution.
b. Develop a design concept for your solution to include a visual diagram of the data structure, process flowcharts and wireframes. (K4, S8, S15, B7)
c. Outline how your design concept meets the relevant guidelines, standards, legalities and regulations. (B5)

Pass criteria	**Distinction criteria**	**Reference to learning** The learning towards this criteria can be supported within the following syllabus areas:
Demonstrate an understanding and creativity to design a fit for purpose solution using appropriate techniques. Communicate information and the solution in a suitable way. Demonstrate consideration and adhere to relevant guidelines, standards, legalities and regulations.	Propose and compare multiple design options, justifying the choice of design.	BCS Level 4 Module in Digital Core: 3.4, 3.6, 5.4 BCS Level 4 Module in Software Developer: 1.4, 6.3, 9.1, 9.2, 9.3

4. Code development

You should provide evidence of how you have developed your solution and your use of particular programming languages and software. This should include samples of your code with explanation to how it has been used to deliver the intended solution and specific functionality.

a. Provide an outline of the development approach you are going to take. (S11)
b. Use the technical specification to build the application including the interface and backend using appropriate, secure, and well-structured code. (S1, S2, S3, S8, S12, S16, S17, B1, B2)
c. Demonstrate the method for version and source control. (S14)
d. Deploy the application into a staging environment ready for testing. (S10)

Pass criteria	**Distinction criteria**	**Reference to learning** The learning towards this criteria can be supported within the following syllabus areas:
Apply clear and valid reasoning in order to create software which is logical, effective and uses appropriate algorithms and data. Identify and apply a suitable development approach. Demonstrate an ability to use a programming language to make an application work.	Demonstrate an ability to develop code that can be reused, explaining how it could be used in multiple areas of the same application.	BCS Level 4 Module in Digital Core: 6.1, 6.2, 6.4 BCS Level 4 Module in Software Developer: 7.2, 7.3, 7.5, 7.6, 7.7, 7.8, 7.9, 7.10, 7.11, 7.13

5. Testing

You will provide evidence of how you have tested your solution, the process you have followed and your observations when testing. This may take the form of a testing log. You should explain your approach to fixing or addressing any issues encountered whilst testing. You will be required to:

a. Select and apply a suitable testing approach, following a stated methodology or framework.
b. Undertake the testing against identified test scenarios. (S4, S5, S6, S13)
c. Provide evidence of how you have identified and resolved problems during testing. (S6,S7, B6, B8)

You should include a copy of your test log in your annex.

Pass criteria	**Distinction criteria**	**Reference to learning** The learning towards this criteria can be supported within the following syllabus areas:
Select and apply a suitable testing approach, following a stated methodology or framework. Proactively identify and resolve problems, using algorithms where required.	Facilitate a pilot or group testing session in order to gather feedback from multiple test users.	BCS Level 4 Module in Digital Core: 6.1, 6.2, 6.4 BCS Level 4 Module in Software Developer: 7.1, 7.2, 7.3, 7.5, 7.6, 7.7, 7.8, 7.9, 7.10, 7.11, 7.13

6. Conclusion and further improvements

You will provide a summary of your solution, evaluating whether it has met all the intended requirements and identifying potential improvements you have identified through testing.

You should include a reflective account, evaluating your performance and any lessons learnt from the process. You will be required to:

a. Review the outputs of the work, identifying potential improvements. (B1, B3, B8, B9)
b. Review own performance, identifying areas for personal development. (B1, B3, B8, B9, B10)

Pass criteria	**Distinction criteria**	**Reference to learning** The learning towards this criteria can be supported within the following syllabus areas:
Review own performance and the output of the work, identifying potential improvements.	Analyse the feedback provided by multiple test users, incorporating their feedback into any modifications or plans for improvement.	BCS Level 4 Module in Digital Core: 7.2 BCS Level 4 Module in Software Developer: 1.3, 2.3, 3.1, 3.3, 3.4

Occupational standard

The following table lists each of the knowledge, skills and behaviours as defined within the Software Developer occupational standard, as referenced in the sections above.

Knowledge
K1: All stages of the software development life-cycle (what each stage contains, including the inputs and outputs)
K2: Roles and responsibilities within the software development lifecycle (who is responsible for what)
K3: The roles and responsibilities of the project life-cycle within your organisation, and your role
K4: How best to communicate using the different communication methods and how to adapt appropriately to different audiences
K5: The similarities and differences between different software development methodologies, such as agile and waterfall
K6: How teams work effectively to produce software and how to contribute appropriately
K7: Software design approaches and patterns, to identify reusable solutions to commonly occurring problems
K8: Organisational policies and procedures relating to the tasks being undertaken, and when to follow them. For example the storage and treatment of GDPR sensitive data
K9: Algorithms, logic and data structures relevant to software development for example:- arrays- stacks- queues- linked lists- trees- graphs- hash tables- sorting algorithms- searching algorithms- critical sections and race conditions
K10: Principles and uses of relational and non-relational databases
K11: Software designs and functional or technical specifications
K12: Software testing frameworks and methodologies
Skills
S1: Create logical and maintainable code
S2: Develop effective user interfaces
S3: Link code to data sets
S4: Test code and analyse results to correct errors found using unit testing
S5: Conduct a range of test types, such as Integration, System, User Acceptance, Non-Functional, Performance and Security testing
S6: Identify and create test scenarios
S7: Apply structured techniques to problem solving, debug code and understand the structure of programmes in order to identify and resolve issues
S8: Create simple software designs to effectively communicate understanding of the program
S9: Create analysis artefacts, such as use cases and/or user stories
S10: Build, manage and deploy code into the relevant environment
S11: Apply an appropriate software development approach according to the relevant paradigm (for example object oriented, event driven or procedural)
S12: Follow software designs and functional or technical specifications
S13: Follow testing frameworks and methodologies
S14: Follow company, team or client approaches to continuous integration, version and source control
S15: Communicate software solutions and ideas to technical and non-technical stakeholders
S16: Apply algorithms, logic and data structures
S17: Interpret and implement a given design whist remaining compliant with security and maintainability requirements

Behaviours
B1: Works independently and takes responsibility. For example, has a disciplined and responsible approach to risk and stays motivated and committed when facing challenges
B2: Applies logical thinking. For example, uses clear and valid reasoning when making decisions related to undertaking work instructions
B3: Maintains a productive, professional and secure working environment
B4: Works collaboratively with a wide range of people in different roles, internally and externally, with a positive attitude to inclusion & diversity
B5: Acts with integrity with respect to ethical, legal and regulatory ensuring the protection of personal data, safety and security
B6: Shows initiative and takes responsibility for solving problems within their own remit, being resourceful when faced with a problem to solve
B7: Communicates effectively in a variety of situations to both a technical and non-technical audience
B8: Shows curiosity to the business context in which the solution will be used, displaying an inquisitive approach to solving the problem. This includes the curiosity to explore new opportunities, techniques and the tenacity to improve methods and maximise performance of the solution and creativity in their approach to solutions
B9: Committed to continued professional development

Top tips you help you prepare

Finally, here are our key pieces of advice for preparing your project:

1. Plan your time wisely – e.g. you should approach this piece of work as if it were a real-life project.
2. Structure your project using the criteria on pages 6-11 of this guidance document.
3. Ensure you adhere to the word limit.
4. Document what you do – e.g. things will fail, but this does not mean that they are not worth writing about. For example, you might like to discuss why they failed. What did you learn? How did this have an impact on your next steps?
5. Use tools that you are familiar with – e.g. do not be tempted to use new or unfamiliar tools for the project, as this could waste time.
6. Show your thought process – e.g. think about how you have used logic or problem-solving techniques to approach and break down a problem. Make sure you document this!
7. Remember to map in an appendix how and where you have evidenced the criteria listed on pages 7-12 of this guidance document. This will not count towards your project word count.
8. Use the project checklist to make sure you have completed everything that is required before submitting your project.

You can find more information in the Software Developer syllabus or the DMP Qualification Guide.

Good luck!