BCS Digital Industries Apprenticeship

Standard Specific Guidance for Training Providers

Level 3 Software Development Technician Apprenticeship

Version 5.1
May 2019

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
Version 1.0 May 2017	Document Created.
Version 2.0 March 2018	Updated competencies and minimum requirements.
Version 3.0 May 2018	Updated work activities and removal of Typical Evidence.
Version 4.0 February 2019	Updates to proficiencies Business Skills, Complexity, Autonomy and Influence throughout the document
Version 5.0 May 2019	Complete document layout overhaul. Competencies and proficiencies unchanged.
Version 5.1 May 2019	Edits to significantly higher competencies, page 29. Edit to significantly higher criteria proficiencies, page 35.

Contents

Purpose of this Document

The purpose of this document is to provide useful information and suggested supporting documentation specific to the Software Development Technician apprenticeship. It should be read in conjunction with the Standard, Occupational Brief and Assessment Plan and is designed to give training providers some tools to help them build their own programme from training plan through to end point assessment (EPA).

This guide will provide supporting information around how to help the apprentice to meet and go beyond the standard and a number of useful documents to support the training provider in meeting their responsibilities in managing the apprenticeship from training plan through to the EPA.

Introduction

The BCS Level 3 Software Development Technician Apprenticeship is one of the suite of Digital Industries Apprenticeships that have been designed by the industry to address skills shortages and meet the ever-changing needs of UK employers.

The BCS website provides the broad view on how to run an apprenticeship programme to the BCS Digital Industries Standard. This document has been designed to give training providers the tools to build their programme and to assist them in helping apprentices and employers towards the successful completion of each element of the EPA.

The areas where a training provider should be involved in ensuring a successful outcome to the apprenticeship are:

- mapping and assessing work against the standard;
- advising the employer and the apprentice on which knowledge modules, vendor or professional certificates and other relevant training and activities are most appropriate for their requirements, and agree a suitable training plan;
- assisting the apprentice with applying knowledge in the workplace;
- acting as an advisor to the apprentice and the employer to ensure the programme remains on track and any concerns are addressed;
- helping the apprentice to select evidence for their summative portfolio;
- supporting the apprentice through the synoptic project;
- confirming the apprentice's readiness for the EPA.

The following series of checklists can be used by the training provider to help manage the process through to completion. Training providers may substitute their own processes and documentation as they see fit in order to effectively manage their key areas of responsibility as set out above.

Standard Specific Guidance for Training Providers – Software Development Technician
V5.1 May 2019

The Software Development Technician Apprentice

A Software Development Technician typically works as part of a software development team, to builds simple software components (whether web, mobile or desktop applications) to be used by other members of the team as part of larger software development projects. They will interpret simple design requirements for discrete components of the project under supervision. The approach will typically include implementing code, which other team members have developed, to produce the required component. . The Software Development Technician will also be engaged in testing that the specific component meets its intended functionality.

Job titles may be different across different organisations so the role may also be referred to as Software Development Technician, Junior Developer, Junior Web Developer, Junior Application Developer, Junior Mobile App Developer, Junior Games Developer, Junior Software Developer, Junior Application Support Analyst, Junior Programmer, Assistant Programmer and Automated Test Developer.

Knowledge Standards, Technical Competence and Behaviour and Relationship Standards

Tables 1, 2 and 3 contain details of the topics that the training provider may decide to cover in their development plans and scheduled work activities in order to stretch the apprentice.

Table 1 – Software Development Technician – Knowledge Standards

The knowledge standards define learning that must take place during the apprenticeship, **both through the activities and the apprentice's own independent learning**. The additional assessment criteria detailed in the table show how a training provider can stretch the apprentice's learning beyond the requirement as set out in the occupational brief. However, it is important to remember that stretching the apprentice in this way will only have a bearing on their final grading if the impact is demonstrated through their competence in the EPA. These knowledge standards, therefore, show the additional learning that may support the apprentice in improving their overall competence.

Technical knowledge and understanding are assessed throughout the apprenticeship through a combination of Ofqual regulated knowledge modules and/or specified vendor and professional qualifications which must be passed before the EPA can take place.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
BCS Level 3 Certificate in Software Development Context and Methodologies	Understand the business context and market environment for software development.	- Knows that software development is undertaken across many industries, but that whilst the processes and methods are broadly similar the data and rationale can be very different. - Understand that it is important to keep digital processes up to date and web systems responsive to user needs. - Understand that some business are virtual web based enterprises, whilst some use web and digital services to engage with clients and customers.	Understand how similar software development processes and methods are used across a range of industries, but can be based on very different rationale. - data. <hr>Identify the factors that may lead to the development of different information systems within or across a range of industry sectors, including. - business requirements; - project timescales; - budgets; - resources and skills availability; - product and project risks. <hr>Explain why businesses need to keep digital processes up to date and web systems responsive to user needs.<hr>Explain the difference between virtual web based enterprises and companies that use web and digital services with respect to customer and client engagement.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
	Understand the structure of software applications.	• Understands the underlying architecture of software applications. • Understands the importance of linking software to databases to store new data inputs collected through applications and to present information choices to users.	Identify the different components that contribute to the underlying architecture of software applications. • code and libraries; • data; • application components; • application interfaces; o network and hardware platforms • reference to the OSI (Open Systems Interconnection) model.
			Describe the use of data sources in software applications for storage and retrieval of information.
	Understands all stages of the software development lifecycle.	• Understands that software is developed through various phases referred to as the software development lifecycle (SDLC). • Understands the main phases of the software development lifecycle and the main activities in each stage.	Recognise that there are several ways to represent the terminology and phases of the SDLC. • feasibility study; • requirements analysis; • design; • code development; • testing; • deployment / implementation; • maintenance.
			Summarise the phases of the SDLC.
			Identify the main activities of each of the phases of the SDLC in terms of inputs, activities and outputs.
			Recognise the relationship between the phases of the SDLC and the roles within the software development team.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
	Understand the role of configuration management and version control systems and how to apply them.	• Understand that code development needs to be managed and controlled using configuration management tools to store incremental developments of code as it is being developed. • Understand that version control manages updates to code and also development of software for different platforms.	Explain how configuration management tools and techniques are used to control and manage the different software development artefacts through the phases of the SDLC and live operation, including: • requirements documentation; • code; • test scripts.
			Summarise the main features and benefits of version control for the development of code including: • change history; • concurrent working; • tracking and preventing conflicts; • traceability; • security.
			Explain how version control can be used for software and software artefacts. • that are being developed for use on multiple platforms; • where similar but slightly different versions need to be produced.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
	Understand how to test their code (e.g. unit testing).	• Understand the different types of software testing (e.g. unit testing, integration testing, load testing etc.) and how to apply them.	Recognise why testing is necessary, including principles of: • early testing; • risk reduction; • conformance to functional and non-functional requirements; • finding and reporting defects; • the difference between testing and debugging.
			Summarise the different levels of testing within the SDLC. • unit; • integration; • system; • acceptance.
			Describe how unit testing follows the fundamental test process consisting of: • test planning, monitoring and control, including maintaining traceability between requirements and testing artefacts; • test analysis and design; • test implementation and execution; • evaluating exit criteria and reporting.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
			Identify the different types and techniques for software testing that are available and why they would be used, including: • functional testing; • non-functional testing; o security o performance o reliability • reviews and static analysis; • white box testing (structure-based); • black box testing (specification-based).
			Recognise the tool types used to support software testing and their main purpose. • test management; • static testing; • test execution; • performance / load / stress testing.
	Recognise that there are different methodologies that can be used for software development.	• Understand that there are a wide range of variations of software development methodologies including waterfall and agile based methods and how to apply them.	Identify the main features of sequential development methods and approaches. • Waterfall.
			Identify the main features of iterative (incremental) development methods and approaches. • Agile.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
			Distinguish between the use of different software development methodologies and approaches, considering their suitability and application to the project. • Agile; • Waterfall.
	Understand the particular context for the development platform (whether web, mobile, or desktop applications).	• Understand that software development needs to reflect the platform the software will be deployed onto (e.g. web, mobile or desktop) and that software may be for a single platform or increasingly multiple platforms.	Explain the features of the following platforms in context of software development, deployment and underlying architecture. • web; • desktop; • mobile; • server; • cloud.
			Distinguish the characteristics of software development that are impacted by the deployment of software. • on multiple platforms; • to a single platform.
	Understands their role within their software development team.		Describe the main roles within software development teams. • requirements engineer; • business analyst; • software designer; • software developer; • software tester; • software project manager; • software release engineer.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
		• Understand the various roles that exist within software development teams and how these relate to each other (e.g. business analyst, requirements engineer, software designer, software developer, software tester, software project manager, software release engineer etc.)	Distinguish how the different roles (as listed in 3.1 [above]) relate / work with each other and their key accountabilities, in order to complete specific activities and tasks.
			Recognise the key external roles and processes that interface to the roles within the software development team (as listed in 3.1 [above]). • customers; • end-users; • operation's processes and personnel; • service management processes and personnel.
			Recognise that collaborative approaches are especially important in Agile development and DevOps practices.
BCS Level 3 Certificate in Programming	Understand how to implement code following a logical approach.		Explain the fundamental concepts of programming. • procedural vs. object-oriented vs. functional programming; • compiled vs. interpreted.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…

		• Understand and apply the fundamental principles and concept of software coding, including abstraction, logic, algorithms and data representation.
		• Understand how to write software code in order to solve problems.
		
		• Understand how to create and debug programs.
			Demonstrate the core constructs used when writing code.
			
			• classes;
			• objects;
			• methods;
			• variables;
			• logic operators;
			    ○ AND
			    ○ OR
			    ○ NOT
			    ○ NAND
			    ○ NOR
			    ○ XOR
			• control structures.
			    ○ iteration
			    ○ selection
			    ○ sequence.

			Explain and demonstrate how algorithms are used. • encryption; • searching; • sorting.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…

			Explain and demonstrate how data structures are used and how data is represented in software code.
			
			• types of data;
			    o integer
			    o floating
			    o Boolean
			    o character
			    o string
			• variables;
			• lists, stacks, arrays.

			Describe how to write software code in order to solve problems.
			
			• describe how programs are structured;
			    o instructions;
			    o sub-routines;
			    o pseudocode;
			    o data definitions and links;
			    o comments;
			• describe modularity and the rational re-use of code;
			    o design patterns;
			    o library functions;
			    o frameworks.

			Understand the fundamental concept of Test Driven Development (TDD).

Standard Specific Guidance for Training Providers – Software Development Technician
V5.1 May 2019

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
	Understand how their code integrates into the wider project.	• Understand the roles and activities needed at each stage of software development. • Understand how software components are managed and controlled and how these are brought together into software solutions. • Understand what team-working aspects are needed to ensure effective delivery of software projects.	Describe the activities undertaken in the following stages of software development: • design; • code development; • testing. Outline the activities undertaken in the following stages of software development: • feasibility study; • requirements analysis; • deployment / implementation. Understand software development activities for the following roles: • requirements engineer; • business analyst; • software designer; • software developer; • software tester; • software release engineer.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…

			Describe the key business concepts and artefacts that must be considered during a software development project.
			• processes and procedures;
			   ○ business process management as it relates to business involvement in development
			   ○ release management
			• documentation;
			• training;
			• support;
			• service levels.

			Describe how software development is conducted within governance structures and the role of the project manager.
			Understand how effective team-working contributes to the effective delivery of software projects. • decision making; • conflict resolution; • collaboration; • communication; • peer review and retrospectives.
	Understand how to follow a set of		Understand how to follow a set of functional and non-functional requirements.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
	functional and non-functional requirements.	• Understand the differences between functional and non-functional requirements and how these are used to drive software development activities. • Understand how to review requirements and consider the testability of each requirement.	Understand the difference between functional and non-functional requirements and how these are used to drive software development activities. • how to review requirements; • how to assess their validity; • how they are used as input to software design; • how they are used during testing to ensure adequate test coverage.
			Identify the different types of non-functional requirements, and the reasons they are important to the end-product of software development. • availability; • capacity; • performance; • scalability; • reliability; • maintainability.
			Recognise common ways in which software requirements can be expressed. • requirements documents – clear, unambiguous; • user stories; • use case diagrams; • process models / flow diagrams; • UML diagrams.
			Describe the qualities of good requirements and the impact of poor requirements.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
			Explain how to determine the correct level of test coverage based on each requirement / type of requirement.
	Understand the end user context for the software development activity.	• Understand the needs of the user and the environment that the software will be used in (e.g. by a doctor in a hospital, or by a consumer through a web-site, or by an engineer in a manufacturing plant).	Understand and recognise the relationship between the user and the environment in which the software will be used.
			Understand the individual business and external constraints and dependencies that need to be taken into account when developing software. • compliance; • ethics; • governance; • legality.
			Describe the methods used to identify end-user needs. • questionnaires; • user interviews; • contextual enquiry; • focus groups; • personas; • customer journey mapping.
	Understand how to connect their code to specified data sources.	• Understand the importance of seamlessly connecting applications to databases that can be used to: ○ store new information (e.g. orders or customer information) ○ extract and displayed stored data (e.g. on products, pricing etc.)	Explain the purpose of data storage for storing new information (orders or customer information). • orders; • customer information.
			Explain the purpose of data storage for extracting and displaying data. • products; • pricing.

Standard Specific Guidance for Training Providers – Software Development Technician
V5.1 May 2019

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
			Explain the concept and key features of databases and data stores. • relational databases; • SQL and NoSQL; • data files; • data structures (tables, records, fields, definitions); • document; • key-value.
	Demonstrate knowledge of database normalisation.	• Understand that normalisation is a method of reducing the complexity of multiple database tables into smaller and well-structured relations. • Understand that a key principle of normalisation is to ensure that information or data should be stored only once.	Explain the purpose and importance of effective data modelling and normalisation.
			Demonstrate the principle of normalisation, that information or data should be stored only once.

	Understand why there is a need to follow good coding practices.	• Understand that good coding practices aid the efficiency and quality of coding development.
		• Understand that there are a range of published open standards and organisational standards, where to
			Explain the importance of good coding practice.
			• quality of coding development.
			  o design documentation
			  o structure of code
			  o consistent design and structure
			  o secure code

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
		source these and how to apply them.	Explain the purpose of good software coding principles and practices. • the basic common principles; o DRY (don't repeat yourself) • defensive programming; • commenting; • refactoring; • patterns / anti-patterns.
			Understand that there are a range of open and organisational coding standards and where to source them.
	Understand the principles of good interface design.	• Understand the issues associated with designing and developing interactive systems. • Understand the main techniques and technologies used for interface design. • Appreciate the importance of usability when developing interactive systems.	Explain human computer interaction and understand the issues associated with interactive systems. • usability / ease of use and intuitive design; • graphical user interfaces (GUI) for different types of devices; • ergonomic design.

			Describe the key concepts and processes of good user interface design.
			• design principles;
			• design patterns;
			• tools;
			   o wireframes;
			   o prototypes;
			• techniques and methods;
			   o A/B testing.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
			Explain the importance of usability when developing interactive systems.
			Describe the fundamental considerations for developing an accessible system and the purpose of the Web Accessibility Initiative (WAI).

	Understand the importance of building in security to software at the development stage.	• Understand what is meant by "building security in" and that this includes:
		  ○ the key steps in the design process to identify and incorporate security requirements into software development.
		  ○ the key techniques for defensive programming such as input checking.
			Describe the following types of security issues and the scale and nature of threats that can impact software development.
			  • common security attacks;
			  • security versus resilience;
			  • social engineering.

			Explain what is meant by 'building security in', in terms of secure software development and creating a secure end-product, and why it is important.
			  • the role coders play in determining a secure software end-product;
			  • the impact they can have on security by not building security in;
			  • why building security in at the start is better than trying to retrofit later.

			Describe proactive security approaches during software design and development.
			  • security development lifecycle (SDLC);
			  • defensive design / defensive programming;
			  • test creation and execution;
			  • permission setting and role based access;
			  • physical infrastructure and security.

Qualification Name	IFA Knowledge Standard	Occupational Brief Expected Minimum Requirement	Assessment Criteria The Learner Can…
			Explain approaches to make software more secure. • security scanning; • penetration testing; • fuzzing; • load testing.

Table 2 – Software Development Technician – Technical Competency Standards

The competency standards have been defined to demonstrate that the knowledge learnt has been applied in real work tasks, activities and projects in a business environment. Competencies are assessed throughout the apprenticeship through a combination of the employer reference, the synoptic project and a summative portfolio completed by apprentices from records of the work activities in which they have been involved. The training provider should assist the employer to identify suitable work tasks, activities and projects within the scope of their normal business activities for the apprentice to practice what they have learnt and to demonstrate all the competencies below.

The BCS apprenticeship is mapped to an internationally recognised skills framework and to work activities in which the apprentice would be involved. The following tables set out these competencies and the expected requirements against the work activities that might be demonstrated at and beyond the minimum expectation:

Competency Standard (IfATE Standard)	Expected Requirement (Occupational Brief)	Work Activities Demonstrating Expected Level of Competence
Logic: writes simple code for discrete software components following an appropriate logical approach to agreed standards (whether web, mobile or desktop applications).	Apprentices can write code to achieve the desired functionality using the appropriate tools and methods applicable to their organisation.	Designs moderately complex programs and program modifications from supplied specifications, using agreed standards and tools, to achieve a well-engineered result. Documents all work in accordance with agreed standards.
Security: applies appropriate secure development principles to specific software components at all stages of development.	Apprentices can apply security principals to all work within the development lifecycle. Apprentices understand the importance of up to date software.	Applies procedures to assess compliance of hardware and software configurations to policies, standards, legal and regulatory requirements.

Competency Standard (IfATE Standard)	Expected Requirement (Occupational Brief)	Work Activities Demonstrating Expected Level of Competence
Development support: applies industry standard approaches for configuration management and version control to manage code during build and release.	Apprentices understand the business context and drivers during development. Apprentices can communicate with customers, internal and external, to explain their work with the appropriate language relevant to their audience. Apprentices understand the importance of version control at every stage within the development lifecycle.	Uses the appropriate operating systems, hardware, tools and/or paper documents to maintain the configuration management system, including the configuration management database (CMDB). Ensures that necessary data, forms and configuration items (CIs) are available for use by all authorised personnel. Assists in the configuration of software and equipment for the systems testing of platform specific versions of one or more software products.
Data: makes simple connections between code and defined data sources as specified.	Apprentices can link to a range of database types and embed data queries within their code.	Works with clients/users on development projects to make effective use of (object) database management systems (O/DBMS), query languages, other DB tools and techniques. Interprets installation standards to meet particular project needs and produces database components as required. Determines what information is required, and specifies search criteria. Plans, designs and conducts tests of programs; corrects errors and re-tests to achieve an error-free result.
Test: functionally test that the deliverables have been met or not.	Apprentices can test and analyse their code to identify errors as soon as possible in the coding process and on an interactive basis. Apprentices can design manual tests for their product. Apprentices understand expected results and acceptance criteria during testing	Reviews requirements and specifications, and defines test conditions. Analyses test requirements, designs and builds simple test case suites, test scripts, and test procedures, with expected results. Checks test results, and documents test failures and successes compared with pre-determined criteria, in accordance with agreed standards.

Competency Standard (IfATE Standard)	Expected Requirement (Occupational Brief)	Work Activities Demonstrating Expected Level of Competence
Analysis: follows basic analysis models such as use cases and process maps.	Apprentices can read and understand data to ensure they know and can meet the customers' requirements. Apprentices can identify and represent required functionality (e.g. use cases). Apprentices can identify and represent activity workflow (e.g. process maps).	Gains awareness of the context of object and data management within the IT function and within the employer's business.
Development lifecycle: supports the Software Developers at the build and test stages of the software development lifecycle.	Apprentices can work as part of a team that understand their process within the development lifecycle. Apprentices can show initiative during development and take responsibility for their own work. Apprentices can work flexible to assist with other member of the team during development	Assists as part of a team on design of components of larger systems. Creates, amends and keeps track of programs in accordance with the design.
Quality: follows organisational and industry good coding practices (including for naming, commenting etc.)	Apprentices can identify and follow standards and good practice that can improve programming efficiency, style and quality, including: - programming standards, both organisational and external; - generic best practices including readability, reusability, maintainability; - best practice approaches of different paradigms and language.	Documents all work using required standards, methods and tools, including prototyping tools where appropriate. Designs simple programs and program modifications from supplied specifications, using agreed standards and tools, to achieve a well-engineered result.

Competency Standard (IfATE Standard)	Expected Requirement (Occupational Brief)	Work Activities Demonstrating Expected Level of Competence
Problem solving: • solves logical problems, seeking assistance when required (including appropriate mathematical application); • responds to the business environment and business issues related to software development.	Apprentices can use at least one problem solving tool and technique to identify and resolve programming issues. Apprentices can apply structured problem solving methods. Apprentices can apply problem-solving techniques to programming activities. Apprentices can demonstrate they know how and where to seek assistance dependent at what stage of development is appropriate.	Assists users in making more effective use of desktop systems, products and services. Makes initial diagnosis of any problems and advises known solutions where applicable.
Communication: clearly articulate the role and function of software components to a variety of stakeholders (including end users, supervisors etc.)	The apprentice should be able to use a minimum of 3 tools to communicate: • Oral; • Face-to-face; • Remote; • Diagrammatic. The apprentice should be able to document work done in accordance with agreed procedures. The apprentice must be able to explain 3 types of communication styles to ensure cultural awareness and appropriateness for customer is taken into account.	Provides guidance and assistance to less experienced colleagues in the execution of routine tasks and ensures that all safety, security, clerical and administrative procedures are completed correctly. Typically, by telephone, email, or instant messaging, provides customers with routine information on the features, operational requirements, products and services supported. Documents work using the required standards, methods and tools.
Operates effectively in their own business's, their customers' and the industry's environments.	Apprentices can demonstrate working within operational requirements such as health and safety, budgets, brands and normal business protocols.	No defined work activities, see expected requirement.

Competency Standard (IfATE Standard)	Expected Requirement (Occupational Brief)	Work Activities Demonstrating Expected Level of Competence
User interface: develops user interfaces as appropriate to the organisations development standards and the type of software development being developed.	Apprentices understand the fundamental concepts of human-computer interaction or user experience design, the development practices leading to a high-quality user interface, and the programming techniques required to construct a graphical user interface. Apprentices understand how to interact with screen or UI designers to ensure the logic layer integrates with the user interface. Apprentices understand how to interact with testers to optimise the user interface.	Designs simple applications using templates and tools to specify user/system interfaces, including for example: menus, screen dialogues, wireframes, boned rigs, inputs, reports, validation and error correction procedures, and processing rules.

Below are the criteria for demonstrating if the apprentice is working at a significantly higher level than the expected level of competence:

Criteria for Demonstrating Significantly Higher Competencies.	Key Indicators
Understands and applies a wide range of tools and methods.	This must be in addition to the range of tools required for a pass and demonstrate solid breadth and depth of knowledge, application and purpose of the tools used.
Accurately and appropriately applies and effectively implements the right tools and methods in a variety of different situations.	These situations / tasks must show a wide range and breadth of situations and be in addition to normal day to day work
A sophisticated user - fully exploits the functionality/capability of the tools and methods.	This must demonstrate solid breadth and depth of functionality, application and purpose of the tools selected. That they have researched and understood the rational for use and not just taken directions from others in the selection.
Extensive and deep understanding of different tools and methods and how and why they can be applied in different contexts.	This must demonstrate breadth and depth of the tools selected, why they have been selected and their appropriateness for the different tasks and uses.
Deals confidently and capably with a high level of interrelated and interdependent factors in their work.	This must demonstrate a confident and consistent approach to all areas of their work (both mundane and interesting work). They should have a thorough understanding and appreciation of their reliance and actions on others work.

Table 3 – Generic Behaviour and Relationship Standards

The behaviour and relationship standards have been defined to demonstrate that the apprentice applies the good behaviours and interpersonal skills that are needed in a business environment. Behaviours and business relationship skills are assessed throughout the apprenticeship through a combination of the employer reference, the synoptic project and a summative portfolio, which is completed by apprentices from records of the work activities in which they have been involved. The training provider could assist the apprentice by offering some additional soft skills training over and above their apprenticeship. The apprenticeship standard sets out the attributes required within the occupation brief, which can be accessed via the Apprenticeship section of www.bcs.org.

Behaviour and Relationship Standard	Expected Requirement
Apprentices can demonstrate the full range of skills, knowledge and behaviours required to fulfil their job role.	Knows what skills, knowledge and behaviours are needed to do the job well. Are aware of their own strengths in the job role, and any areas for improvement. Appreciate who else is important, for them to do their job and fulfil the role effectively (e.g. colleagues, managers, other stakeholders). Are aware of potential risks in the job role (e.g. security, privacy, regulatory). Use personal attributes effectively in the role. Understand how the job fits into the organisation as a whole.
Apprentices can demonstrate how they contribute to the wider business objectives and show an understanding of the wider business environments.	Understands the goals, vision and values of the organisation. Aware of the commercial objectives of the tasks/ projects they are working on. Understands the importance of meeting or exceeding customers' requirements and expectations. Is in tune with the organisation's culture. Aware of the position and contribution of the organisation in the economy. Understands the key external factors that shape the way the organisation function, e.g. regulation. Knows how the organisation can gain advantage in the industry, e.g. through innovation, technology, customer service etc.

Behaviour and Relationship Standard	Expected Requirement
Apprentices can demonstrate the ability to use both logical and creative thinking skills when undertaking work tasks, recognising and applying techniques from both.	Logical thinking: • Understands initial premise(s) and preconditions; • Recognises the conclusion to be reached; • Proceeds by rational steps; • Evaluates information, judging its relevance and value; • Supports conclusions, using reasoned arguments and evidence. Creative thinking: • Explores ideas and possibilities; • Makes connections between different aspects; • Adapts ideas and approaches as conditions or circumstances change.
Apprentices can show that they recognise problems inherent in, or emerging during, work tasks, and can tackle them effectively.	Problem-solving: • Analyses situations; • Defines goals; • Develops solutions; • Prioritises actions; • Deals with unexpected occurrences.

Behaviour and Relationship Standard	Expected Requirement
Apprentices can manage relationships with work colleagues, including those in more senior roles, customers / clients and other stakeholders, internal or external, and as appropriate to their roles, so as to gain their confidence, keep them involved and maintain their support for the task / project in hand. Apprentices can establish and maintain productive working relationships, and can use a range of different techniques for doing so.	Managing relationships: - Understands the value and importance of good relationships; - Influences others by listening to and incorporating their ideas and views; - Acknowledges other people's accomplishments and strengths; - Manages conflict constructively; - Promotes teamwork by encouraging others to participate; Customer/client relationships: - Understands their requirements, including constraints and limiting factors; - Sets reasonable expectations; - Involves them in decisions and actions; - Interacts positively with them; - Provides a complete answer in response to queries ('transparency', 'full disclosure') Stakeholders: - Understands who they are and what their 'stake' is; - Prioritises stakeholders in terms of their importance, power to affect the task and interest in it; - Uses stakeholders' views to shape projects early on; - Gains support from stakeholders, e.g. to win resources; - Agrees objectives.

Behaviour and Relationship Standard	Expected Requirement
Apprentices can communicate effectively with a range of people at work, one-to-one and in groups, in different situations and using a variety of methods. Apprentices can demonstrate various methods of communication, with an understanding of the strengths, weaknesses and limitations of these, the factors that may disrupt it, and the importance of checking other people's understanding.	Intention/purpose: • Understands the purpose of communicating in a particular situation or circumstance (e.g. inform, instruct, suggest, discuss, negotiate etc.); • Checks that the person/people with whom one is communicating also understand the purpose; • Is sensitive to the dynamics of the situation; • Is aware of anything that might disrupt the effectiveness of the communication (e.g. status, past history); Method: • Chooses a good, appropriate method for the situation; • Aware of the limitations of the chosen method, and the possible risks of miscommunication (e.g. ambiguity); • Takes account of the affective dimensions of the method (e.g. body language, tone of voice, eye contact, facial expression etc.); Execution: • Expresses self clearly and succinctly, but not over-simplifying; • Checks that the other person/people understand what is being expressed; • Takes account of the potential barriers to understanding (e.g. filtering, selective perception, information overload); • Modifies the purpose and methods of communication during a situation in response to cues from the other person/people.

These attributes are difficult to measure and are subjective in nature so cannot guarantee that any greater level of competence or proficiency is being demonstrated. The BCS apprenticeship is mapped to the Skills Framework for the Information Age (SFIA), an internationally recognised skills framework and to observable activities that an apprentice working to the level of responsibility appropriate for the role should demonstrate. Accordingly, the proficiencies that should be demonstrated by the apprentice are shown below.

Proficiency Standard	Work Activities Demonstrating Expected Level of Competence
Business skills	Demonstrates an analytical and systematic approach to issue resolution. Demonstrates effective communication skills. Contributes fully to the work of teams. Appreciates the wider business context, and how their role relates to other roles and to the business of the employer or client.
Complexity	Performs a range of work, sometimes complex and non-routine, in a variety of environments. Applies a methodical approach to issue definition and resolution. Undertakes all work in accordance with agreed safety, technical and quality standards, using appropriate methods and tools.
Influence	Has working level contact with customers, suppliers and partners.
Autonomy	Works under general direction. Determines when issues should be escalated to a higher level.

Below are the criteria for demonstrating if the apprentice is working at a significantly higher level than the expected level of proficiency:

Proficiency Standard	Work Activities Demonstrating Competence Beyond the Minimum Expected
Business skills	Works independently and takes high level of responsibility.
	Undertakes work that is more complex, more critical or more difficult.
	Independently demonstrates an ability to extend or enhance their approach to work and the quality of outcomes.
	Doesn't just solve the problem but explores creative or innovative options to do it better, more efficiently, more elegantly or to better meet customer needs.
	Shows strong project management skills, in defining problem, identifying solutions and making them happen.
Complexity	Demonstrates a disciplined approach to execution, harnessing resources effectively.
	Drives solutions – with a strong goal focused and appropriate level of urgency.
Influence	Externally – works with customers, suppliers, and partners in a variety of situations.
	Actively inspires and leads others, takes others with them, leads by example.
Autonomy	Internally – works alone, 1:1, in a team and across the company with colleagues at all levels.
	Reads situation, adapts behaviours, and communicates appropriately for the situation and the audience.
	Can be trusted to deliver, perform and behave professionally, manages and delivers against expectations, proactively updates colleagues and behaves in line with the highest values and business ethics.

Software Development Technician Apprentice Templates

The following templates are designed to support the training provider, and will take them from training and development planning, through to the EPA readiness check. As with the tables above they can be used by the training provider to help them manage the process through to completion, although training providers may also substitute their own processes and documentation as they see fit in order to effectively manage their programme.

Template 1 – Training and Development Plan

Apprentice Details

Name	
ULN number	

Employer Details

Contact name	
Company name	
Company address	

Training Provider Details

Contact name	
Company name	
Company address	

Role Mapping Against the Software Development Technician Standard

For each area of technical and behavioural competence an overall evaluation should be provided on a three-point scale to show how often this competence is required during the normal work carried out by the employer:

- competence is applied most of the time;
- competence is applied some of the time;
- competence is rarely required.

This evaluation could form the basis of an ongoing review with the apprentice on a regular basis.

Workplace Competence Map

This template shows the type of activities that are identified in the apprenticeship standard.

It is recognised that there are differences between the types of work carried out by different employers, so this template provides the opportunity to include any other activity that demonstrates the apprentice's competence during their normal duties.

The tables below could be used to make an evaluation of the apprentice's work environment and detail the work activities that a competent apprentice should be able to undertake. This activity should then lead to a discussion to identify any gaps with the employer and make a plan to redress the balance.

Competency Standard	Is the apprentice required to demonstrate the competency in the normal course of work?		
	Most of the Time	Some of the Time	Rarely
Logic: writes simple code for discrete software components following an appropriate logical approach to agreed standards (whether web, mobile or desktop applications).	☐	☐	☐
Security: applies appropriate secure development principles to specific software components at all stages of development.	☐	☐	☐
Development support: applies industry standard approaches for configuration management and version control to manage code during build and release.	☐	☐	☐
Data: makes simple connections between code and defined data sources as specified.	☐	☐	☐
Test: functionally test that the deliverables have been met or not.	☐	☐	☐
Analysis: follows basic analysis models such as use cases and process maps.	☐	☐	☐
Development lifecycle: supports the Software Developers at the build and test stages of the software development lifecycle.	☐	☐	☐

Competency Standard	Is the apprentice required to demonstrate the competency in the normal course of work?		
	Most of the Time	Some of the Time	Rarely
Quality: follows organisational and industry good coding practices (including for naming, commenting etc.)	☐	☐	☐
Problem solving: • solves logical problems, seeking assistance when required (including appropriate mathematical application); • responds to the business environment and business issues related to software development.	☐	☐	☐
Communication: clearly articulate the role and function of software components to a variety of stakeholders (including end users, supervisors etc.)	☐	☐	☐
Operates effectively in their own business's, their customers' and the industry's environments.	☐	☐	☐
User interface: develops user interfaces as appropriate to the organisation's development standards and the type of software development being developed.	☐	☐	☐

What is your overall evaluation of the apprentice's opportunity to demonstrate the technical competencies in the employer's normal workplace environment?

Please continue on a separate sheet if required.

Knowledge Module Training Plan

The knowledge standards define learning that should take place during the apprenticeship, both through the training provider activities and the apprentice's independent learning. The training provider should work with the employer to identify appropriate training for the apprentice to meet the requirements of the standard and the employer should identify opportunities within the scope of their normal business activities for the apprentice to demonstrate what they have learnt.

Knowledge and understanding will be delivered through BCS qualifications in accordance with the standard.

Training Plan – Knowledge

BCS Qualification	Completed
BCS Level 3 Certificate in Software Development Context and Methodologies	☐
BCS Level 3 Certificate in Programming	☐

Technical Competence Development Plan

The following template may be used to ensure that the apprentice will be given the opportunity to demonstrate each of the required technical competencies stated in the standard.

Responsibility	Employer ☐	Training Provider ☐
Logic: writes simple code for discrete software components following an appropriate logical approach to agreed standards (whether web, mobile or desktop applications).		
How will this be ensured?		

Responsibility	Employer ☐	Training Provider ☐
Security: applies appropriate secure development principles to specific software components at all stages of development.		
How will this be ensured?		

Responsibility	Employer ☐	Training Provider ☐
Development support: applies industry standard approaches for configuration management and version control to manage code during build and release.		
How will this be ensured?		

Standard Specific Guidance for Training Providers – Software Development Technician
V5.1 May 2019

	Responsibility	Employer	☐	Training Provider	☐

Data: makes simple connections between code and defined data sources as specified.

How will this be ensured?

	Responsibility	Employer	☐	Training Provider	☐

Test: functionally test that the deliverables have been met or not.

How will this be ensured?

	Responsibility	Employer	☐	Training Provider	☐

Analysis: follows basic analysis models such as use cases and process maps.

How will this be ensured?

	Responsibility	Employer ☐	Training Provider ☐

Development lifecycle: supports the Software Developers at the build and test stages of the software development lifecycle.

How will this be ensured?

	Responsibility	Employer ☐	Training Provider ☐

Quality: follows organisational and industry good coding practices (including for naming, commenting etc.)

How will this be ensured?

	Responsibility	Employer ☐	Training Provider ☐

Problem solving:
- solves logical problems,seeking assistance when required (including appropriate mathematical application);
- responds to the business environment and business issues related to software development.

How will this be ensured?

	Responsibility	Employer ☐	Training Provider ☐

Communication: clearly articulate the role and function of software components to a variety of stakeholders (including end users, supervisors etc.)

How will this be ensured?

	Responsibility	Employer ☐	Training Provider ☐

Operates effectively in their own business's, their customers' and the industry's environments.

How will this be ensured?

	Responsibility	Employer ☐	Training Provider ☐

User interface: develops user interfaces as appropriate to the organisations development standards and the type of software development being developed.

How will this be ensured?

Template 2 – Weekly Diary

Week number	Activities completed	Competencies displayed	Supporting evidence

Template 3 – Periodic Workplace Competence Assessment and Remedial Action Plan

This template can be used to track the competencies being applied in the workplace on a continual / periodic basis. The training provider can then discuss any gaps with the employer and make a plan to redress the balance.

Competence assessment

Is the apprentice meeting the minimum competence standard?	☐

Logic: writes simple code for discrete software components following an appropriate logical approach to agreed standards (whether web, mobile or desktop applications).
What should the apprentice start, stop or continue doing in order to develop this competence?

Is the apprentice meeting the minimum competence standard?	☐

Security: applies appropriate secure development principles to specific software components at all stages of development.
What should the apprentice start, stop or continue doing in order to develop this competence?

Is the apprentice meeting the minimum competence standard?	☐

Development support: applies industry standard approaches for configuration management and version control to manage code during build and release.
What should the apprentice start, stop or continue doing in order to develop this competence?

Standard Specific Guidance for Training Providers – Software Development Technician
V5.1 May 2019

Is the apprentice meeting the minimum competence standard?	☐

Data: makes simple connections between code and defined data sources as specified.

What should the apprentice start, stop or continue doing in order to develop this competence?

Is the apprentice meeting the minimum competence standard?	☐

Test: functionally test that the deliverables have been met or not.

What should the apprentice start, stop or continue doing in order to develop this competence?

Is the apprentice meeting the minimum competence standard?	☐

Analysis: follows basic analysis models such as use cases and process maps.

What should the apprentice start, stop or continue doing in order to develop this competence?

Is the apprentice meeting the minimum competence standard?	☐

Development lifecycle: supports the Software Developers at the build and test stages of the software development lifecycle.

What should the apprentice start, stop or continue doing in order to develop this competence?

Is the apprentice meeting the minimum competence standard?	☐

Quality: follows organisational and industry good coding practices (including for naming, commenting etc.)

What should the apprentice start, stop or continue doing in order to develop this competence?

Is the apprentice meeting the minimum competence standard?	☐

Problem solving:
- solves logical problems,seeking assistance when required (including appropriate mathematical application);
- responds to the business environment and business issues related to software development.

What should the apprentice start, stop or continue doing in order to develop this competence?

Is the apprentice meeting the minimum competence standard?	☐

Communication: clearly articulate the role and function of software components to a variety of stakeholders (including end users, supervisors etc.)

What should the apprentice start, stop or continue doing in order to develop this competence?

Is the apprentice meeting the minimum competence standard?	☐

Operates effectively in their own business's, their customers' and the industry's environments.

What should the apprentice start, stop or continue doing in order to develop this competence?

Is the apprentice meeting the minimum competence standard?	☐

User interface: develops user interfaces as appropriate to the organisations development standards and the type of software development being developed.

What should the apprentice start, stop or continue doing in order to develop this competence?

Remedial action plan

An important function of the training provider is to act as an advisor to the apprentice and the employer to ensure that the programme remains on track and any concerns are addressed. The training provider should agree how best to provide ongoing assistance / advice throughout the apprenticeship, possibly as part of their contract / service agreement with the apprentice's employer.

If any remedial action is required, the table below could be used to record it.

Please continue on a separate sheet as required.

Information contained within this document has been republished under the terms of the Open Government Licence v3.0 © Crown copyright (2019)
Copyright © BCS 2019
Skills Framework for the Information Age © SFIA Foundation 2003, 2005, 2008, 2011, 2015
SFIA*plus* © The British Computer Society 2004, 2006, 2008, 2011, 2015
Standard Specific Guidance for Training Providers – Software Development Technician
V5.1 May 2019

Page **50** of **59**

Template 4 – The Employer Reference

Overview

This template and guidance will assist the training provider in supporting the employer when completing the employer reference, which forms a key part of the EPA. The intent of the employer reference is for the employer to support the apprentice by validating the evidence that they have submitted for EPA.

The employer will be asked to provide an overall evaluation of the apprentice for each area of technical competence and behavioural proficiency, giving detail of how the apprentice meets each requirement.

This guidance shows the type of activities that could demonstrate the required competencies and behaviours being applied in the workplace. There are always differences between individual employers and their requirements so there is the opportunity for the employer to include any other activity that they think demonstrates the apprentice's competence. It should be completed by a senior member of the team, who is able to comment directly on work activities.

The apprenticeship standards are designed to cover a wide range of different job roles so there may be a small number of areas within these mandatory requirements that are not naturally occurring within the day-to-day duties of the apprentice. If it is not possible for the apprentice to demonstrate competence within their duties, a synoptic project should be selected that will allow the apprentice to demonstrate that they are competent in criteria that they are not exposed to during their normal working activities.

The template is provided as a standalone editable document and can be found on the BCS Accredited Provider area. This should be completed by the employer and submitted for review as part of the EPA.

Template 5 – Summative Portfolio Checklist

This template will support the training provider in working with the apprentice and employer to ensure the successful completion of the summative portfolio.

The checklists can be used by training providers to help them manage the process through to completion, although training providers may also substitute their own processes and documentation as they see fit.

The apprentice should gather artefacts and record information that can evidence their activities undertaken in the workplace. The portfolio of evidence should demonstrate that the apprentice can fulfil the full range of competencies which are required by the standard, as shown in this template.

The apprenticeship standards are designed to cover a wide range of different job roles so there may be a small number of areas within these mandatory requirements that are not naturally occurring within the day-to-day duties of the apprentice. If it is not possible for the apprentice to demonstrate competence within their summative portfolio, a synoptic project should be selected that will allow the apprentice to demonstrate that they are competent in criteria that they are not exposed to during their normal working activities.

The template is provided as a standalone editable document and can be found on the BCS Accredited Provider area.

Template 6 – EPA Readiness Check

This template is to support the training provider in assessing whether the apprentice has met the criteria for the EPA, as defined in the standard.

Is the apprentice ready?	☐

Logic: writes simple code for discrete software components following an appropriate logical approach to agreed standards (whether web, mobile or desktop applications).

Comments

Is the apprentice ready?	☐

Security: applies appropriate secure development principles to specific software components at all stages of development.

Comments

Is the apprentice ready?	☐

Development support: applies industry standard approaches for configuration management and version control to manage code during build and release.

Comments

Standard Specific Guidance for Training Providers – Software Development Technician
V5.1 May 2019

	Is the apprentice ready?	☐

Data: makes simple connections between code and defined data sources as specified.

Comments

	Is the apprentice ready?	☐

Test: functionally test that the deliverables have been met or not.

Comments

	Is the apprentice ready?	☐

Analysis: follows basic analysis models such as use cases and process maps.

Comments

	Is the apprentice ready?	☐

Development lifecycle: supports the Software Developers at the build and test stages of the software development lifecycle.

Comments

	Is the apprentice ready?	☐

Quality: follows organisational and industry good coding practices (including for naming, commenting etc.)

Comments

	Is the apprentice ready?	☐

Problem solving:
- solves logical problems,seeking assistance when required (including appropriate mathematical application);
- responds to the business environment and business issues related to software development.

Comments

Standard Specific Guidance for Training Providers – Software Development Technician

V5.1 May 2019

Is the apprentice ready?	☐

Communication: clearly articulate the role and function of software components to a variety of stakeholders (including end users, supervisors etc.)

Comments

Is the apprentice ready?	☐

Operates effectively in their own business's, their customers' and the industry's environments.

Comments

Is the apprentice ready?	☐

User interface: develops user interfaces as appropriate to the organisations development standards and the type of software development being developed.

Comments

Professional Development

Activities Plan

BCS has defined a number of professional development activities that support wider professional and career development. These activities have been associated with the various levels of responsibility, and the activities listed in the table below represent those that are appropriate for an apprentice.

Training providers may wish to engage in assisting the apprentice in some of these activities as they can contribute towards the portfolio of evidence. The recommended activities include those shown below.

Professional Development Activities	Appropriate to the Role	Agreed with Apprentice and Employer
Participating in group activities inside or outside the working environment that can assist with the development of interpersonal skills.	☐	☐
Undertaking unpaid activities that can help to develop professional skills or offer additional insight into, or understanding of, their working role.	☐	☐
Undertaking learning in subjects relevant to, but not directly related to, their role (e.g. mentoring skills, cultural awareness and diversity training), perhaps through self-study or evening classes.	☐	☐
Gaining basic knowledge of the employing organisation, its business, structure, culture, products/services, operations and terminology.	☐	☐
Gaining knowledge of IT activities in the employing organisation external to their function.	☐	☐
Exploring a topic that is not part of their normal responsibilities, and presenting findings to colleagues and/or management.	☐	☐
Attending meetings, seminars and workshops organised by a professional body, and reading published material such as journals and web content.	☐	☐
Undertaking learning and practice in the techniques of team and collaborative working. Gaining an understanding of the underlying concepts.	☐	☐
Undertaking learning and practice in oral and written communications, including report writing and presentations.	☐	☐

Activities Typical Evidence

Areas of additional professional development activities that might be undertaken and associated typical evidence are shown below.

Professional Development Topic	Objectives	Typical Evidence
Understanding organisation	Gaining basic knowledge of the employing organisation, its business, structure, culture, products/services, operations and terminology. Gaining knowledge of IT activities in the employing organisation external to their function.	• organisation charts; • company annual reports; • company website; • documents or reports from other areas of the business.
Additional business skills	Undertaking learning and practice in the techniques of team and collaborative working. Gaining an understanding of the underlying concepts. Undertaking learning and practice in oral and written communications, including report writing and presentations. Learning from experience and mistakes and applying the lessons as part of continuous improvement.	• presentations, reports or minutes of meetings that demonstrate communication skills, report writing abilities and collaborative activities; • evidence of reviewing their work and suggesting improvements or critically appraising what they did and what they learned from it.
External activities	Participating in group activities inside or outside the working environment that can assist with the development of interpersonal skills. Undertaking pro bono (unpaid) activities that can help to develop professional skills or offer additional insight into, or understanding of, their working role.	• evidence of meetings attended through continuous professional development records; • evidence of activities undertaken.

Professional Development Topic	Objectives	Typical Evidence
Additional learning	Undertaking learning in subjects relevant to, but not directly related to, their role (e.g. foreign language courses, mentoring skills, cultural awareness and diversity training), perhaps through self-study or evening classes. Exploring a topic that is not part of their normal responsibilities, and presenting findings to colleagues and/or management.	• evidence of learning undertaken from continuous professional development records; • evidence of presentations given to colleagues and/or management.
Professional networking	Attending meetings, seminars and workshops organised by a professional body and reading published material such as journals and web content.	• evidence of meetings attended through continuous professional development records; • written evidence summarising learning gained from reading.