Software Development Technician Apprenticeship Standard

Role/Occupation: Software Development Technician

Level 3

Overview: Building simple applications for use in larger software developments.

A Software Development Technician typically works as part of a software development team, to build 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.

DURATION

The apprenticeship will typically take 21 months to complete.

ENTRY REQUIREMENTS

Entry requirements will be determined by individual employers.

ENGLISH & MATHS

Apprentices without Level 2 English or Maths will need to achieve this prior to taking their End Point Assessment.

For those with an education, health and care plan or a legacy statement, the minimum English and Maths requirement is Entry Level 3. For those whose primary language is British Sign Language, BSL qualifications are an alternative to English qualifications.

QUALIFICATIONS

There are no mandatory qualifications for this apprenticeship standard.

COMPETENCIES

Knowledge	What is required
K1	Fundamentals of all stages of the software development life cycle including development, Quality Assurance, User Acceptance Testing and release
K2	Roles and responsibilities within the software development life-cycle
K3	Roles and responsibilities of the project life–cycle
K4	Different communication methods, how to adapt appropriately to different audiences including collaborative technologies like discussion threads and document collaboration
K5	The key similarities and differences between different software development methodologies, such as agile and waterfall
K6	Principles of effective teamwork to produce software

Knowledge	**What is required** *(Continued)*
S4 K7 S5	Fundamentals of software design approaches and patterns, including when 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 General Data Protection Regulation (GDPR) sensitive data
K9	Fundamentals of computing systems including physical, virtual and cloud technologies
K10	Fundamental principles of algorithms, logic and data structures. For example, how they work using a step-by-step solution to a problem, or rules to follow to solve the problem
K11	Principles and uses of relational and non-relational (NoSQL) databases
K12	Basic principles of software designs and functional/ technical specifications
K13	Key principles of software testing frameworks and methodologies
K14	Principles of pattern recognition such as looking for similarities among and within problems
K15	Fundamentals of breaking down a complex problem or system into smaller, more manageable parts
K16	The importance of valuing difference and understanding the protected characteristics named in the Equality Act 2010
K17	Basic principles of emerging technology trends and innovations such as Internet of Things (IoT) Artificial Intelligence (AI) Augmented Reality (AR)
K18	Awareness of legal and regulatory requirements and their practical application to the role for example, Data Protection, Security, Intellectual Property Rights (IPR), Data sharing, marketing consent, personal data definition
K19	Fundamental approaches to actions such as sequence, selection and iteration
K20	Basic principles of software project planning including: Risks and dependencies, integration, prioritisation of tasks, escalation of problems, quality, time, end user experience
K21	Basic principles of processes and protocols used to ensure internet security, including concepts of security assurance.
K22	Key principles of testing for components (including software, hardware, data), interfaces and the resulting service
K23	Role and importance of Industry Standards and where to find them (e.g., ISO standards, IETF RFCs)
K24	Software development approaches for example object oriented, event driven or procedural Standards

Skill	What is required
S1	Write simple code for discrete software components following an appropriate logical approach to agreed standards (web/mobile/desktop applications) under supervision
S2	Apply appropriate secure development principles to specific software components at all stages of development
S3	Support development of effective user interfaces
S4	Make simple connections between code and defined data sources as specified
S5	Test simple code & analyse results to correct errors found using unit testing under supervision
S6	Conduct a range of test types under supervision, such as Functional and Non-Functional

Skills	**What is required** *(Continued)*
S7	Apply structured techniques to problem solving, including carry out simple code debug
S8	Follows organisational and industry good coding practices (including for naming, commenting etc.)
S9	Solve logical problems, seeking assistance when required (including appropriate mathematical application)
S10	Support the creation of simple software documentation and visuals to effectively communicate understanding of the program
S11	Define functional and non -functional requirements such as use cases, storyboards, user stories, performance and accessibility
S12	Work within operational requirements such as health and safety, budgets, brands and normal business protocols
S13	Develop user interfaces as appropriate to the organisation's development standards and the type of software development being developed
S14	Build scripts in line with work instructions for deployment into the relevant environment
S15	Follow simple software designs and functional/technical specifications in line with work
S16	Follow simple testing frameworks and methodologies in line with work instructions
S17	Follow company, team or client approaches to continuous integration, version and source control as instructed
S18	Support the communication of software solutions and ideas to technical and non-technical stakeholders
S19	Apply algorithms, logic and data structures in a supported context in line with work instructions
S20	Follow work instructions to contribute to building a given design whist remaining compliant with security and maintainability requirements
S21	Apply techniques to break down complex problems
S22	Demonstrate how Key Performance Indicators (KPIs) can be used to frame and measure desired outcomes
S23	Implement secure code in appropriate languages of different types which is maintainable, readable, functional
S24	Design simple software solutions to meet a requirement using tools and techniques, such as waterfall and agile
S25	Work in a shared code base with appropriate etiquette and tools, such as modularity and data definition
S26	Use simple debugging techniques, such as interactive debugging, print debugging, remote debugging
S27	Implement test plans under supervision to show that a test plan is implementable in practice and implementation conforms to the plan
S28	Develop and use simple acceptance criteria
S29	Apply and maintain procedures and security controls to ensure confidentiality, integrity and availability
S30	Use collaboration tools and technologies for source and version control to enable working together on common projects, regardless of physical location
S31	Follow instructions to ensure client data is held securely under supervision e.g., not using personally identifiable information in test systems, making sure personal actions comply with ICO regulations

Skill	What is required *(Continued)*
S32	Use collaboration tools and technologies for writing technical documentation for, and adapting to, specific audience(s), e.g., technical, non-technical, internal, external

Behaviour	What is required
B1	Use critical thinking skills when undertaking work tasks
B2	Committed to guided Continuous Professional Development
B3	Work independently and take responsibility within tightly defined parameters
B4	Maintain a productive, professional and secure working environment
B5	Team player, for example working collaboratively, keeping others informed using effective communication, recognising personal and professional limitations and seeking advice when necessary
B6	Self-motivated, for example manages own time effectively, takes responsibility to complete the job

ON-PROGRAMME DELIVERY

As part of our delivery, apprentices will be invited to attend Marketing Workshops to underpin the knowledge elements of the apprenticeship, as well as other relevant workshops to support the development of the Knowledge, Skills and Behaviours.

Each apprentice will be allocated a Coach to support ongoing learning and preparation for End Point Assessment. In addition Progress Reviews will take place regularly to ensure the apprentice is on track.

END POINT ASSESSMENT

The EPA consists of two assessment methods:

1. Project report with questioning
2. Professional discussion underpinned by portfolio

The EPA will be conducted by an Independent External Assessment Organisation (IEAO).

PROGRESSION

Please talk to us about progression from this apprenticeship.

REALITY CHECK

□ Time and support required from the employer to the apprentice during the apprenticeship to include: regular Performance Reviews, relevant off the job training and preparation for the final EPA

□ Expectation of significant amounts of study/work from the apprentice in order to meet the requirements of the apprenticeship

□ Employer has to be involved in the EPA and provide support and time to the apprentice in preparation for the EPA

COSTINGS

Maximum Funding Band: £15,000

The cost of the apprenticeship will be negotiated with you in line with Government guidelines.

PLEASE CONTACT APPRENTICESHIPS@CIRENCESTER.AC.UK FOR FURTHER INFORMATION
TELEPHONE: 01285 626259