Qualification Specification

Accelerate People L4 EPA for DevOps Engineer ST0825/AP02

QAN: 610/2605/0

Version History		
Version	**Date Amended**	**Changes Made**
1.0	November 2022	Document Created
1.1	April 2023	Edits to wording throughout to focus on assessment criteria. QAN added.
1.2	November 2024	Document rebranded. Guidance amended inline with assessment plan update changing wording from 'typical' to 6-month' for resit timeframe (page 20).
2.0	February 2025	Edited gateway requirements to accommodate change in policy for English and Maths requirements.

Contents

Qualification Objective

The level 4 DevOps Engineer apprenticeship is one of a suite of apprenticeships that have been designed by industry employers to meet a range of job roles across different industries and sizes of business.

Accelerate People are an end-point assessment organisation (EPAO) for the digital apprenticeship standards that are defined by the Institute for Apprenticeships & Technical Education (IfATE). The apprenticeship standard and assessment plan can be found on the IfATE website.

As part of this apprenticeship all apprentices are required to complete an independent end-point assessment (EPA). The purpose of the EPA is to independently assess that any apprentice on this standard is competent in a relevant job role and can evidence meeting all the assessment criteria relating to the knowledge, skills and behaviours (KSB) outcomes.

The Level 4 DevOps Engineer Apprenticeship

Role Profile:

This occupation is found in a wide and diverse range of public and private sector organisations, from tech start-ups through government departments to multinationals. Essentially any organisation of any size that builds and/or operates modern IT services.

The broad purpose of the occupation is to enable organisations to get valuable working software out in front of active users, both external and internal, frequently and safely, reducing time to market, delivering increased value - both with respect to the end user and the business - and improving the quality of digital services. At its simplest, DevOps is a philosophy and way of working that brings together two historically disparate parts of the IT organisation, namely those who develop the software and those who are then required to support it in the live environment. The DevOps Engineer encapsulates both disciplines, requiring the individual to understand and appreciate how their code functions when being used in the real world and troubleshoot any issues that may arise, while taking a cloud-infrastructure focused perspective. This means taking responsibility for all aspects of the development and operations process - the design, build, test, implement, release and continual iteration of products. Utilizing the advantages of Cloud computing to enable infrastructure to be defined in code moves the operations side away from traditional system administrator roles which are focused

on troubleshooting traditional infrastructure-as-hardware. The convergence of these two topics drives DevOps culture and ways of working and creates the need for the new role of DevOps Engineer that works within the delivery team. The DevOps Engineer applies all the DevOps culture and software engineering disciplines to codified infrastructure.

In their daily work, an employee in this occupation interacts with other members of agile development teams, other areas within the organisation's IT department and business areas, as well as 3rd-party suppliers. This is an office based or remote working role, with co-location preferable.

An employee in this occupation will be responsible for working collaboratively with a minimum of direct supervision within broad but generally well-defined parameters, requiring the application of knowledge and understanding, skills and methods in a broad range of complex or technical work activities, performed in a variety of contexts. They will address problems which are non-routine while normally fairly well defined, taking responsibility for courses of action, including, where relevant, responsibility for the work of others and allocation of resources. They will typically pair with other technical roles, or work alone at times, and provide input to the planning of work and advise on design.

Typical job titles:

Automation Engineer, Build and Release Engineer, Deployment Engineer, DevOps Engineer, Full Stack Developer, Infrastructure Engineer, Platform Engineer, Reliability Engineer, Site Reliability Engineer.

Duties:

This apprenticeship standard includes duties to support alignment between the job role and the apprenticeship standard. Listed below are the duties that all apprentices must demonstrate in their apprenticeship. These duties are not assessed or graded as part of the EPA.

Duty 1: Script and code in at least one general purpose language and at least one domain-specific language to orchestrate infrastructure, follow test driven development and ensure appropriate test coverage.

Duty 2: Initiate and facilitate knowledge sharing and technical collaboration with teams and individuals, with a focus on supporting development of team members.

Duty 3: Engage in productive pair/mob programming to underpin the practice of peer review.

Duty 4: Work as part of an agile team, and explore new ways of working, rapidly responding to changing user needs and with a relentless focus on the user experience. Understand the importance of continual improvement within a blameless culture.

Duty 5: Build and operate a Continuous Integration (CI) capability, employing version control of source code and related artefacts.

Duty 6: Implement and improve release automation & orchestration, often using Application Programming Interfaces (API), as part of a continuous delivery and continuous deployment pipeline, ensuring that team(s) are able to deploy new code rapidly and safely.

Duty 7: Provision cloud infrastructure using APIs, continually improve infrastructure-as-code, considering use of industry leading technologies as they become available (e.g., Serverless, Containers).

Duty 8: Evolve and define architecture, utilising the knowledge and experience of the team to design in an optimal user experience, scalability, security, high availability and optimal performance.

Duty 9: Apply leading security practices throughout the Software Development Lifecycle (SDLC).

Duty 10: Implement a good coverage of monitoring (metrics, logs), ensuring that alerts are visible, tuneable and actionable.

Duty 11: Keep up with cutting edge by committing to continual training and development - utilise web resources for self-learning; horizon scanning; active membership of professional bodies such as Meetup Groups; subscribe to relevant publications.

Duty 12: Look to automate any manual tasks that are repeated, often using APIs.

Duty 13: Accept ownership of changes; embody the DevOps culture of 'you build it, you run it', with a relentless focus on the user experience.

Entry Requirements

Qualifications

Apprentices aged 16-18 on their apprenticeship start date, without level 2 English and maths, will need to achieve this level prior to taking the EPA. For those with an education, health and care plan or a legacy statement, the apprenticeship's English and maths minimum requirement is Entry Level 3. A British Sign Language (BSL) qualification is an alternative to the English qualification for those whose primary language is BSL.

Apprentices aged 19+ on their apprenticeship start date, without level 2 English and maths, are exempt from achieving this prior to taking their EPA; this exemption is by prior agreement between the apprentice and employer.

Experience

There are no pre-requisite knowledge, skills or understanding requirements defined for entry onto this qualification.

EPA Requirements

To successfully complete the level 4 DevOps Engineer apprenticeship apprentices must achieve at least a pass in each EPA assessment method. This EPA consists of two discrete assessment methods which have the following grades awarded.

- Assessment Method 1 (AM1): Project and practical assessment.
    - Fail.
    - Pass.
    - Distinction.
- Assessment Method 2 (AM2): Professional discussion.
    - Fail.
    - Pass.
    - Distinction.

All assessment methods must be taken within a four-month period, otherwise the entire EPA will need to be re-sat/re-taken.

EPA Gateway

For this apprenticeship all apprentices must spend a minimum of 12 months on programme, of which a minimum of 20% must be spent undertaking off-the-job training, before being eligible to undertake the EPA.

Before starting the EPA, an apprentice must meet the following gateway requirements:

- The employer is satisfied that the apprentice is working at or above the occupational standard.
- Apprentices must have achieved English and Maths qualifications in line with the apprenticeship funding rules.

Apprentices may request additional time if they require a reasonable adjustment. Information on how and when to apply is contained within the reasonable adjustments policy.

Once the apprentice is ready to enter gateway the following must be submitted to progress:

- Gateway form:
    - Demonstrating where evidence has met the outcomes listed on the standard.
    - Demonstrating where the knowledge has been completed and uploading evidence of any certificates, if applicable.
    - Confirming the preferred date for each assessment method.
    - Advising Accelerate People if the apprentice requires any reasonable adjustments to be made during the EPA.
    - Confirmation signatures that the apprentice is ready for the EPA.
- Evidence of:
    - Maths and English qualifications at Level 2 or above (or acceptable equivalent as specified in the entry requirements section), **or**
    - Confirmation that the apprentice is exempt from achieving English and Maths qualifications.
- The apprentice's project brief summary (for AM1), see Project for further details.

The gateway form along with all documentation must be uploaded before the EPA can commence. Failure to upload any of the required documentation may delay the EPA start date.

Project Brief Summary

The project brief summary is to be submitted to the EPAO at the gateway:

- The apprentice will scope out and provide a brief summary of what the project will cover and will submit this to the EPAO at the gateway. This should demonstrate that the work-based project will provide sufficient opportunity for the apprentice to develop the piece of code and meet the assessment criteria.
- The brief summary needs to outline the project plan, including high level implementation steps and associated timeframes, as well as the date the work-based project has to be submitted to the independent assessor, taking into account the deadlines stipulated within this end-point assessment plan.
- The brief summary should typically be no more than 500 words and is not an assessed element of the EPA.
- Within two weeks of receiving the project brief the EPAO will the title of the project report.

Knowledge, Skills and Behaviours

There are no mandatory vendor qualifications or knowledge modules for this apprenticeship. Apprentices are expected to be able to demonstrate competence against the assessment criteria specified within the assessment plan. The assessment criteria are based on the following KSBs, which apprentices are expected to be competent in before entering gateway.

Knowledge

K1: Continuous Integration - the benefits of frequent merging of code, the creation of build artefacts and ensuring all tests pass, with automation throughout - including common tooling.

K2: The principles of distributed Source Control, including how to exploit the features of the tool, such as branching.

K3: How to use data ethically and the implications for wider society, with respect to the use of data, automation and artificial intelligence within the context of relevant data protection policy and legislation.

K4: The business value of DevOps in terms of Time, Cost, Quality, with an emphasis on building in internal Quality throughout the lifetime of the product.

K5: A range of modern security tools and techniques - e.g., threat modelling, vulnerability scanning and dependency checking, with a general awareness of penetration testing - in order to deal with threats and attack vectors within code and across the cyber domain.

K6: A range of problem solving techniques appropriate to the task at hand, such as affinity mapping, impact maps, plan-do-check-act/Deming.

K7: General purpose programming and infrastructure-as-code.

K8: Immutable infrastructure and how it enables continuous refreshing of software, namely the updating of the operating system, container and security patching.

K9: Different organisational cultures, the development frameworks utilised and how they can both complement each other and introduce constraints on delivery.

K10: How the user experience sits at the heart of modern development practices in terms of strategies to understand diverse user needs, accessibility and how to drive adoption.

K11: Monitoring and alerting technologies and an awareness of the insights that can be derived from the infrastructure and applications - collecting logs and metrics, configuring alerting thresholds, firing alerts and visualising data.

K12: The persistence/data layer, including which database/storage technologies are appropriate to each platform type and application when considering non-functional and functional needs, e.g. monolith, microservice, read heavy, write heavy, recovery plans.

K13: Automation techniques, such as scripting and use of APIs.

K14: Test Driven Development and the Test Pyramid. How the practice is underpinned by unit testing, the importance of automation, appropriate use of test doubles and mocking strategies, reducing a reliance on end-to-end testing.

K15: The principles and application of Continuous Integration, Continuous Delivery and Continuous Deployment, including the differences between them.

K16: How best to secure data, e.g., encryption in transit, encryption at rest and access control lists (ACL).

K17: What an API is, how to find them and interpret the accompanying documentation.

K18: Roles within a multidisciplinary team and the interfaces with other areas of an organisation.

K19: Different methods of communication and choosing the appropriate one - e.g., face-to-face (synchronous, high bandwidth), instant messaging, email (asynchronous, low bandwidth), visualisations vs. words.

K20: Pair/mob programming techniques and when to use each technique.

K21: Architecture principles, common patterns and common strategies for translating user needs into both cloud infrastructure and application code.

K22: How their occupation fits into the wider digital landscape and any current or future regulatory requirements.

K23: The importance of continual improvement within a blameless culture.

K24: The difference between Software-as-a-Service (SaaS) v bespoke v enterprise tooling and how to make an informed choice that suits each use case.

K25: Maintain an awareness of cloud certification requirements.

Skills

S1: Communicate credibly with technical and non-technical people at all levels, using a range of methods, e.g., 'Show and Tell' and 'Demonstrations'.

S2: Work within different organisational cultures with both internal and external parties.

S3: Translate user needs into deliverable tasks, writing clear, concise and unambiguous user stories that the whole team can understand.

S4: Initiate and facilitate knowledge sharing and technical collaboration.

S5: Deploy immutable infrastructure.

S6: Install, manage and troubleshoot monitoring tools.

S7: Navigate and troubleshoot stateful distributed systems, in order to locate issues across the end-to-end service.

S8: Work in agile, multi-disciplinary delivery teams, taking a flexible, collaborative and pragmatic approach to delivering tasks.

S9: Application of a range of cloud security tools and techniques - e.g., threat modelling, vulnerability scanning, dependency checking, reducing attack surface area - incorporating these tools and techniques into the automated pipeline wherever possible.

S10: Assess identified and potential security threats and take appropriate action based on likelihood v impact.

S11: Employ a systematic approach to solving problems, using logic and hypotheses / experimentation to identify the source of issues.

S12: Automate tasks where it introduces improvements to the efficiency of business processes and reduces waste, considering the effort and cost of automation.

S13: Engage in productive pair/mob programming.

S14: Write tests and follow Test Driven Development discipline in various different contexts.

S15: Release automation and orchestration as part of a Continuous Integration workflow and Continuous Delivery pipeline, automating the delivery of code from source control to the end users.

S16: Invest in continuous learning, both your own development and others, ensuring learning activities dovetail with changing job requirements. Keep up with cutting edge.

S17: Code in a general-purpose programming language.

S18: Specify cloud infrastructure in an infrastructure-as-code domain-specific language.

S19: Interpret logs and metrics data within the appropriate context to identify issues and make informed decisions.

S20: Writing code in such a way that makes merging easier and facilitates branching by abstraction - i.e., feature toggling.

S21: Application of lightweight modelling techniques, such as whiteboarding, in order to gain consensus as a team on evolving architecture.

S22: Incremental refactoring by applying small behaviour-preserving code changes to evolve the architecture.

Behaviours

B1: Exhibits enthusiasm, openness and an aptitude for working as part of a collaborative community, e.g. sharing best practice, pairing with team members, learning from others and engaging in peer review practices.

B2: Invests time and effort in their own development, recognising that technology evolves at a rapid rate.

B3: Displays a commitment to the mantra 'You build it, you run it', taking ownership of deployed code and being accountable for its continual improvement, learning from experience and taking collective responsibility when things fail.

B4: Is inclusive, professional and maintains a blameless culture.

Assessment

AM1: Project with Practical Assessment

The apprentice will be observed by an independent assessor completing a practical assessment during which they will demonstrate the assessment criteria assigned to this assessment method based on a post-gateway work-based project. They will submit an electronic-based project output (i.e., the piece of code) to the EPAO after a maximum of 13 weeks following EPAO sign-off of the project brief. Apprentices will complete the project once they have passed the gateway. Following submission of the project output, the practical assessment will take place with an independent assessor.

Project

Whilst completing the project, the apprentice should be subject to normal workplace supervision.

The following could be included in the piece of code:

- Building a piece of infrastructure and deploying an application to it.
- Building an element of a platform, resident on this infrastructure.
- Development of a new approach to a platform/infrastructure/deployment problem, i.e., novel tooling where no alternates are available.
- Development of tooling to automate common deployment/maintenance processes.
- Development/implementation of new CI/CD pipelines.
- Development of management/support processes.
- This list is not exhaustive.

The apprentice must produce sufficient evidence of the form, technical breadth, and specific technical outputs of the work in order that the independent assessor can familiarise themselves with the project output (i.e., the piece of code) prior to the practical assessment.

As a minimum the submitted project outputs must include:

- For 'form': an architectural diagram (in a structured or ad-hoc notation) or other artefact which shows high level system structure.
- For 'technical breadth': a short analysis, maximum 300 words, of which project areas provide
  - evidence against which KSBs.
- For 'specific technical outputs': the independent assessor will need to be provided with implementations which cover all techniques used. These may include source code, deployment/system build scripts or configuration files and should be communicated to the independent assessor through access to cloud services, an archive of files or in screenshots/videos/documents.

Practical Assessment

Apprentices will demonstrate the piece of code that has been produced following the project and the independent assessor will question them on the methodology used to develop the code.

The practical assessment will cover:

- Operating a performant, secure and highly available platform.
- Satisfy the functional and non-functional requirements defined by the work-based project.
- Meets the assessment criteria mapped to this assessment method.
- A successful deployment of code from source to the end user.

The independent assessor can ask questions throughout the practical assessment and at the end to allow the apprentice to evidence any gaps in assessment criteria not evidenced by the practical assessment. A maximum of 16 questions will be asked. Questions will only be based on the assessment criteria for this assessment method.

Key points:

- The practical assessment will take place online via video conferencing.
- Apprentices will need access to the internet and a working webcam.
- The apprentice must have access to a quiet room and, unless reasonable adjustments have been requested for additional support, be alone in the room.
- Apprentices must have access to a whiteboard for their practical assessment.
- Apprentices must have photographic identification (ID) to verify their identity, if they do not produce any ID then the practical assessment will be cancelled.

- The practical assessment will be carried out over a maximum assessment time of three hours, with the independent assessor having the discretion to increase the time of the questioning by up to 10% to allow the apprentice to complete their last task.
- Up to 16 questions will be asked during and after the practical assessment and will be formed based on the evidence and grading requirements in the table below.
- Apprentices are allowed access to their submitted project outputs throughout the practical assessment.
- Questions will only be based on the evidence submitted for this assessment method.
- Apprentices will have 10 days' notice of the practical assessment date.

AM2: Professional Discussion

The professional discussion will take place at least two weeks after the gateway has been confirmed.

- The professional discussion will take place online via video conferencing.
- Apprentices will need access to the internet and a working webcam for the entire duration.
- The apprentice must have access to a quiet room and, unless reasonable adjustments have been requested for additional support, be alone in the room.
- Apprentices must have photographic identification (ID) to verify their identity, if they do not produce any ID then the professional discussion will be cancelled.
- The discussion will last for 60 minutes with the independent assessor having the discretion to increase the time of the questioning by up to 10% to allow the apprentice to complete their last answer.
- A minimum of 8 questions will be asked and will be formed based on the grading requirements in the table below.
- The apprentice must have access to a whiteboard and may use it to help with visualising the KSBs, e.g. K18, K19 and K22.

Assessment Criteria

AM1

Themes and KSBs	Pass Criteria	Distinction Criteria
Code Quality K2, K5, K7, K14, S9, S11, S14, S17, S18, S20, S22.	Writes code, both general purpose and infrastructure-as-code (including cloud infrastructure) that is correctly versioned and easy to merge, while adhering to the principles of distributed Source Control. Demonstrates an iterative approach to evolving code consistent with cloud security best practice, evidenced by a lack of vulnerabilities and that all dependent components are present at run time. Writes code around unit tests, including the appropriate use of test doubles and mocking strategies. Explains troubleshooting methods used to identify and resolve issues and gives an example of identifying and remediating an issue that compromised code quality.	None specified.
Meeting User Needs K4, K10, K21, S3.	Writes user stories that are understandable to a wide range of stakeholders, stand up to scrutiny and lend themselves to a solution based on common architectural patterns - i.e., reducing the number of moving/redundant parts; passes all acceptance tests. The piece of code meets the 'must have' identified functional/non-functional user needs encapsulated in the acceptance criteria for the task.	Produces a piece of code that meets the 'should have' identified functional/non-functional user needs encapsulated in the acceptance criteria for the task.

Themes and KSBs	Pass Criteria	Distinction Criteria
	Creates a quality product in terms of Mean Time To Recovery (MTTR) - i.e., reduced time to fix bugs.	
The CI-CD Pipeline K1, K15, S15.	Builds a fully functioning, automated CI-CD pipeline with all tests passing. Evidences a code commit progressing seamlessly from a build artefact to the end user. Explains the pipeline capability, including the benefits of frequent merging of code, in terms of Continuous Integration/Delivery/Deployment.	None specified.
Refreshing and Patching K8, S5.	Deploys immutable infrastructure that enables the regular recycling of servers and refreshing of associated software based on manual processes.	Fully automates the refreshing and patching process.
Operability K11, S6, S19, B3.	Installs and manages monitoring and alerting tools that provide coverage of the infrastructure and applications, including RAM and CPU utilisation, application error rates and availability (health check). Configures appropriate alerting thresholds and visualisations. Interprets these in terms of failure scenarios and remedial/follow up actions taken to deliver continuous improvement.	Introduces custom metrics that provide additional improvement areas. Explains how these improvement areas may be interpreted, implemented and delivered.

Themes and KSBs	Pass Criteria	Distinction Criteria
Data Persistence K12, S7.	Employs and operates an appropriate data persistence technology, such as database, configuration/infrastructure state management to meet non-functional and functional needs. Explains troubleshooting steps taken to locate issues across the end-to-end service.	None specified.
Automation K13, K17, S12.	Introduces process efficiencies by automating the setting up/deploying of the project (infrastructure and applications) from scratch, both locally, including all tests, and to a hosted environment.	Identifies an additional opportunity and introduces automation that reduces overall effort.
Data Security K16, S10.	Builds in security so that all data in transit is encrypted and secure. Explains the types of threats and the rationale behind the decision to either encrypt data at rest or not.	None specified.

AM2

Themes and KSBs	Pass Criteria	Distinction Criteria
Organisational Culture K9, K23, S2.	Explains how an organisation's culture can both provide creative freedom and introduce constraints. Explains the connection between culture and the organisation's potential for continuous improvement with	Explains the mindsets that underpin organisational culture - e.g., outcome versus activity driven, collaboration versus silos, accountability, trust and empowerment and their impact on the organisation. Assesses the difference between risk avoidance and risk

Themes and KSBs	Pass Criteria	Distinction Criteria
	both internal and external parties.	acceptance and how these link to culture.
Data Ethics K3.	Identifies relevant data protection legislation and assesses its impact on the ethical use of customer data, as well as its relevance to emerging technologies, such as Artificial Intelligence and Machine Learning.	None specified.
Problem Solving K6, S21.	Identifies different problem-solving techniques and evaluates how they use modelling approaches that are best suited to each technique in order to gain consensus as a team.	Describes how they facilitated an incident post-mortem/lesson learned session. Explains the root cause analysis process. Gains consensus on an improvement plan, including accountabilities and the implementation timeline.
The Profession in Context K18, K19, K22, S8, B4.	Identifies the typical multi-disciplinary team roles and explains how they fit within the organisation and the wider digital landscape. Explains how they completed a task, deploying a flexible, collaborative and pragmatic approach with peers and other stakeholders. Describes examples of different communication methods used when dealing with internal and external stakeholders. Explains how they have acted in an inclusive and professional manner.	None specified.
Tooling & Technology K24.	Explains the difference between the various types of implementation - on premise v	Justifies their choice of tooling and the potential impact of making an alternative choice

Themes and KSBs	Pass Criteria	Distinction Criteria
	SaaS, open-source v enterprise, bespoke v off-the-shelf. Explains an example of having utilised the right type of tool for a particular task, describing the pros and cons of the alternatives.	explaining the cause and effect of making the wrong decision.
Continuous Learning & Development K25, S16, B2.	Explains the CPD undertaken by themselves in order to keep up with cutting edge technologies and maintain appropriate certifications. Explains how they invest in others continuous learning and activities and the impact this has on their own development.	Gives examples of how their CPD has had a positive impact on theirs and their team's work. Explains how this has helped them perform their role better and make better technology choices.
Peer Review K20, S13.	Explains the benefits, in terms of security and overall quality, of subjecting written code to the scrutiny of others. Explains how they collaborate on code through pair/mob commits.	None specified.
Communicating and Knowledge Sharing S1, S4, B1.	Explains when they have: a) Lead a demonstration or discussion in an engaging manner, communicating at the right level to suit technical and non-technical audiences. b) b) Worked collaboratively to share knowledge through, for example, blog posts and pairing on tasks.	None specified.

Grading

Each assessment method is graded individually and combined to give an overall grade. Assessment criteria do not appear in more than one assessment method, therefore an assessment criteria failed in one assessment method cannot then be demonstrated in

the other assessment method. All EPA methods must be passed for the EPA to be passed overall. AM1 is weighted more than AM2, this is reflected in the overall EPA grading table.

Grades from individual assessment methods should be combined in the following way to determine the grade of the EPA as a whole:

Assessment Method 1 – Project and Practical Assessment	Assessment Method 2 – Professional Discussion	Overall Grading
Fail	Fail	Fail
Fail	Pass	Fail
Fail	Distinction	Fail
Pass	Fail	Fail
Pass	Pass	Pass
Pass	Distinction	Pass
Distinction	Fail	Fail
Distinction	Pass	Merit
Distinction	Distinction	Distinction

Re-sits and Re-takes

Apprentices who fail one or more assessment method will be offered the opportunity to take a re-sit or a re-take at the employer's discretion. The apprentice's employer will need to agree that either a re-sit or re-take is an appropriate course of action.

A re-sit does not require further learning, whereas a re-take does. Apprentices should have a supportive action plan to prepare for a re-sit or a re-take.

An apprentice who fails one or more assessment methods, and therefore the EPA in the first instance, will be required to re-sit or re-take the failed assessment method(s) only. The same project/code may be used in the event this method is failed.

Any assessment method re-sit or re-take must be taken during a **6-month period**, otherwise the entire EPA must be taken again, unless in the opinion of the EPAO exceptional circumstances apply outside the control of the apprentice or their employer.

Re-sits and re-takes are not offered to apprentices wishing to move from pass/merit to a higher grade.

Where any assessment method has to be re-sat or re-taken, the apprentice will be awarded a maximum EPA grade of pass, unless the EPAO determines there are exceptional circumstances requiring a re-sit or re-take.

Specimen

All specimen materials, such as an example practical assessment, can be accessed by registered training providers from the knowledge area on ACE360.

Accelerate People

Accelerate People are an independent EPAO specialising in digital apprenticeship EPAs. If you have any questions or queries relating to this qualification specification or EPA, please contact us using the details below.

Registered training providers with Accelerate People can access further guidance material on the knowledge base on ACE360.

Contact Details:

Email: info@accelerate-people.co.uk.

Visit: www.accelerate-people.co.uk Registered training providers with Accelerate People can access further guidance material on the knowledge base on ACE360.

Registered office: Accelerate People Limited, Scale Space, Imperial College White City Campus, 58 Wood Lane, London, W12 7RZ.

Registered in England with number: 09577006.

Registered Apprenticeship Assessment Organisation Number: EPO 0475.