

DevOps Engineer Apprenticeship Standard

Role/Occupation: DevOps Engineer

Level 4



Overview: Enabling organisations to get valuable working software out in front of active users and improving the quality of digital 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.

DURATION

The apprenticeship will typically take 24-28 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:

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
The principles of distributed Source Control, including how to exploit the features of tool, e.g. branching
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
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
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 the threats and attack vectors within code and across the cyber domain
A range of problem solving techniques appropriate to the task at hand, such as affinity mapping impact maps, plan-do-check act/Deming
General purpose programming and infrastructure-as-code
Immutable infrastructure and how it enables continuous refreshing of software, namely the updating of operating system, container and security patching
Different organisational cultures, the development frameworks utilised and how both can complement each other and introduce constraints on delivery
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
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
The persistence/data layer, including which database/storage technologies are appropriate to each platform type and application when considering non-functional/functional needs e.g. monolith, microservice, read heavy, write heavy, recovery plans
Automation techniques, such as scripting and use of APIs
Test Driven Development and Test Pyramid - how practice is underpinned by unit testing, importance of automation, appropriate use of test doubles/mocking strategies, reducing reliance on end-to-end testing
The principles and application of Continuous Integration, Continuous Delivery and Continuous Deployment, including the differences between them
How best to secure data; e.g. encryption in transit, encryption at rest and access control lists (ACL)
What an API is, how to find them and interpret the accompanying documentation
Roles within a multidisciplinary team and the interfaces with other areas of an organisation
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
Pair/mob programming techniques and when to use each technique
Architecture principles, common patterns and common strategies for translating user needs into both cloud infrastructure and application code
How their occupation fits into the wider digital landscape and any current or future regulatory requirements
The importance of continual improvement within a blameless culture
The difference between Software-as-a-Service (SaaS) v bespoke v enterprise tooling & how to make informed choice
Maintain an awareness of cloud certification requirements

Skills - What is required:

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

Work within different organisational cultures with both internal and external parties

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

Initiate and facilitate knowledge sharing and technical collaboration

Deploy immutable infrastructure

Install, manage and troubleshoot monitoring tools

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

Work in agile, multi-disciplinary delivery teams, taking flexible, collaborative & pragmatic approach to task delivery

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

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

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

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

Engage in productive pair/mob programming

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

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

Invest in continuous learning, both your own development and others, ensuring learning activities dovetail with changing job requirements

Code in a general purpose programming language

Specify cloud infrastructure in an infrastructure-as-code domain-specific language

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

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

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

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

Behaviours - What is required:

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

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

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

Is inclusive, professional and maintains a blameless culture

ON-PROGRAMME DELIVERY

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.

All apprentices will be invited to attend relevant workshops to support the development of their Knowledge, Skills and Behaviours.

END POINT ASSESSMENT

The EPA consists of two assessment methods:

1. Project and Practical Assessment
2. Professional Discussion

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 apprenticeships
- 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: £17,000

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

Additional Costs:

- Optional - IT Functional Skills Level 2 - £200