DASA

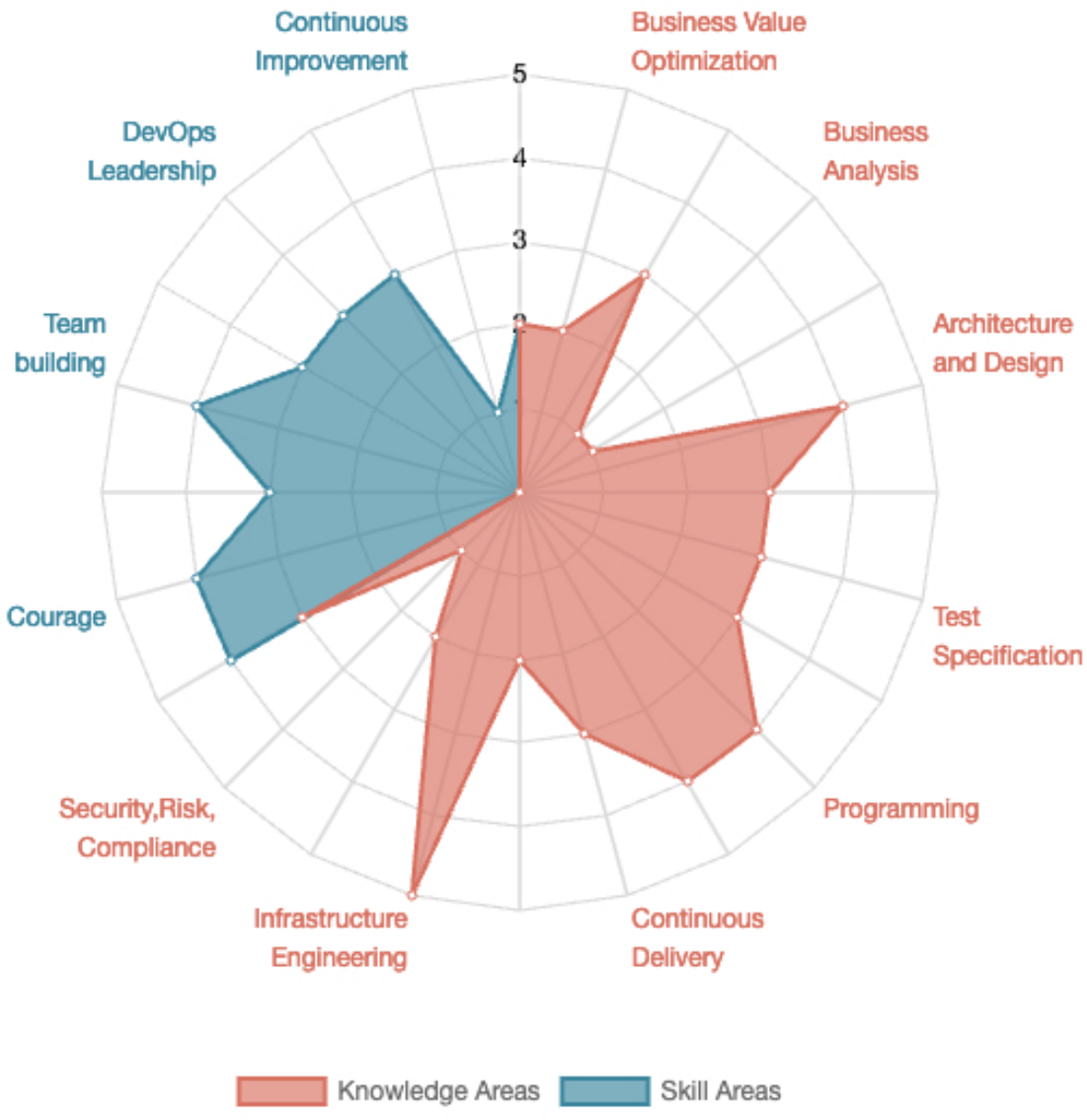
DEVOPS AGILE
SKILLS ASSOCIATION

DASA DEVOPS COMPETENCE QUICKSCAN™

Your report on DASA DevOps competence

YOUR COMPETENCE MODEL

BELOW IS YOUR DETAILED COMPETENCE REPORT



THE COMPETENCE MODEL

HELP ORGANIZATIONS ASSESS THEIR CURRENT DEVOPS COMPETENCY GAP

DASA's competence model reflects the vision that the role of the IT engineer will become more generic as DevOps teams develop. Anyone of the crew can roughly do the job of another. The key to working in this environment is to recognize that there is a skills and knowledge set that needs to be present in every DevOps team. The distribution of these skills and knowledge may be different per team. However, each team will need to ensure that there is enough of each skill and knowledge area to ensure the service is delivered as required by the customers of the service.

1. Novice / 2. Competent / 3. Proficient / 4. Expert / 5. Master

12 SKILLS AND BEHAVIOR AREAS

The DASA competence model identifies 8 knowledge areas and 4 skills and behavior areas that are relevant in DevOps. Every professional operating in a DevOps team requires all 12 competencies in varying degree. This is described in the DASA professional qualification program.

SKILL AREAS

1. **Courage:** Evangelism, coaching, self-confidence, proactivity, reflection, trust, open discussions, experimentation, fail fast, courage to change.
2. **Teambuilding:** Understand the other's point of view, collaboration, mutual accountability, common purpose, ability to integrally support the service/product.
3. **DevOps Leadership:** Facilitating teams to high performance, humility, transparency, Service lifecycle mindset, Stakeholder management.
4. **Continuous improvement:** Today we do our work better than yesterday, kaizen mindset, quality at the source, first time right, knowledge-sharing, ability to adapt.

KNOWLEDGE AREAS

1. **Business Value Optimization:** Use of the IT service in real life, including direct feedback loop of user comments to team, service level management, definition of done, business activity/ performance monitoring, business case management.
2. **Business Analysis:** Functional requirements, non-functional requirements, longer term development of business process (based on translation of market developments), data analysis, and refinement.
3. **Architecture & Design:** Ensuring fit between developments and current situation, overall service design, patterns & styles.
4. **Programming:** Software engineering mastery, everything as code, data management.
5. **Continuous Delivery:** Automated testing, deployment and release management, configuration management, version control, cloud, containerization, feature-driven delivery.
6. **Test Specification:** Design of test cases, test concepts.
7. **Infrastructure Engineering:** Technical monitoring, performance management (e.g load balancing etc.), capacity and availability management, reliability engineering, cloud, containerization.
8. **Security, Risk & Compliance:** Security, service continuity planning.

HOW DOES DEVOPS FUNDAMENTALS FIT INTO THE DASA COMPETENCE FRAMEWORK?

After completing this course, you will cover the area marked as DevOps Fundamentals in the following figure of the DASA qualification scheme. As a result, you will reach the **Competent** level of the scheme.

1. Novice / 2. Competent / 3. Proficient / 4. Expert / 5. Master

HOW DOES DEVOPS PROFESSIONAL – ENABLE AND SCALE FIT INTO THE DASA COMPETENCE FRAMEWORK?

After completing this course, you will cover the area marked as DevOps Professional – Enable and Scale in the following figure of the DASA certification scheme. As a result, you will reach the **Proficient** level of the scheme.

1. Novice / 2. Competent / 3. Proficient / 4. Expert / 5. Master

DASA MATURITY MODEL – SKILL AREAS

		Level 1 – Novice	Level 2 – Competent	Level 3 – Proficient	Level 4 – Expert	Level 5 – Master
COURAGE	Team	Individuals take unconcious risks with little clear understanding of level of risk and possible mitigations. There is little trust between team members.	Individuals understand the consequence of taking risks and resort to strong risk avoidance. Individuals have an initial desire to work differently, but react to signals for change rather than initiate them.	Individuals discuss risks in the team. The foremost reaction is to be cautious. However, taking risks has also provided benefits. There is little knowledge of experimentation techniques. Team members coach each other.	The team takes a proactive stance to (changing) their work. The team agrees on risks to be taken based on analysis of the key aspects of the service delivery. The team has a method by which risks are assessed, and experiments are carried out. The team focuses on failing as fast as possible.	Experimentation is based on agreed risk-reward mechanism that the team has integrated into its way of working. The team is self-confident in its ability to deliver value to its customers through its way of working and proactively shares its experiences and lessons learned.
	Leadership	The reaction of management to unsuccessful risks causes individuals to become more risk-averse. Risk-taking behavior and experimentation are not rewarded.	Management supports this risk avoidance behavior. On occasion, management realizes and communicates that taking risks is necessary to achieve improvements. Management encourages open discussions in the team.	Management stresses the need to take risks and to experiment. However, support for possible undesirable outcomes is not always present. Management helps to build trust in the team.	Management supports and encourages reflection on intended actions. As a result of the use of agreed experimentation techniques, management is more supportive of risk-taking. Management takes an active stance in creating a reward system.	Management encourages risk-taking, understanding both the upside and possible downside of the risk. Management ensures that experiments can run their course.
TEAMBUILDING	Team	Group of individuals without a clear common purpose. No focus on creating common work products or results. Individuals do their best.	Individuals realize that they have reached the limits of what is possible as individuals and seek to identify how they can increase their impact on both their surroundings, the customer and their individual development.	Forming/Storming phase of teambuilding. Individuals find their place in the team, taking on a specific role. Team starts to take ownership of their area of responsibility. Initial signs of accountability.	Norming/Performing phase of teambuilding. Team members become aware of their mutual strengths and weaknesses and optimize the way of working based on this knowledge. Team members encourage learning within the team.	Multi-disciplinary team with all skills required to provide the services required autonomously. Large overlap in skills and knowledge between team members. Team members are able to switch between roles within the team. The team is fully mutally accountable.
	Leadership	Management steers the organization based on individual targets and capabilities.	Management realizes that the sum of the parts is greater than the individuals added together. Management focuses on individual accountability.	Management builds teams around clearly defined common purposes. They select the team members and bring them together to deliver the specific purpose. Leaders are aware that they need to develop high-performance teams.	Leadership encourages the diversity of views in the team to enhance reflection and accelerate innovation. Leaders help to manage conflict constructively. Leaders use team-building best practices to help the team to achieve its goals.	Management explicitly facilitates teams in their improvements. Leaders are focused on steering emerging behaviors instead of actively being involved in team dynamics.

DASA MATURITY MODEL – SKILL AREAS

		Level 1 – Novice	Level 2 – Competent	Level 3 – Proficient	Level 4 – Expert	Level 5 – Master
DEVOPS LEADERSHIP	Team	Individuals look to the manager for their work. They are reactive and take little accountability for their actions.	Individuals occasionally reflect on their tasks and suggest improvements. They are more active in determining what needs to be done.	Team adopts the service lifecycle mindset. The team takes over the technical leadership from the formal hierarchy.	The team takes over leadership of the development and delivery roadmap of the products and services from the formal hierarchy.	Team members fulfil the leadership roles themselves, actively promote their way of working and help other teams and the organization to reflect and improve.
	Leadership	Traditional western leadership, distant from teams and work. Little understanding of the dynamics of the team.	Managers ensure that stakeholders are managed. Managers ensure that time is taken to reflect on individual performance.	Managers retain formal organizational leadership and actively participate in empowering the team to reflect and improve.	Formal hierarchical leadership actively transfers responibility and authority to the teams.	Inspirational and facilitative leadership close to the work floor, aimed at removing impediments beyond the control of the team.
CONTINUOUS IMPROVEMENT	Team	People act in the 'fire-fighting' mode. There is little in the way of structured problem solving and people have no time to think about improvements.	Improvements emerge based on individual initiative and mostly in their own time. These impulses become more frequent as team members see the value in improving.	Improvement time is explicitly included in regular work. Problems are defined and seen in a different light: as the difference between the current and the desired state. There is an agreed structured problem-solving method.	Teams are energized by the search for problems with the express aim of solving them. They recognize the need to redefine the standard thereby creating a new problem to be solved. Teams look outside the boundaries of the team to realize end-to-end solutions for customers.	The team redefines its goals autonomously and continuously in order to drive improvement. Improving the service is a daily activity including both small improvements and solving larger problems using an agreed method that is well-known within the team.
	Leadership	Managers provide the answers. No improvement objectives are defined. Problems are seen as undesirable and are often ignored.	Managers tell teams that they need to improve. There is a very high level plan for improvement without concrete steps and goals. Managers appreciate initiatives but do not ensure that time is made available for improvements.	Managers ensure that time is allocated for improvements and that this time is used for improvements. Managers understand the value of a structured problem-solving method and ensure that both they and the teams know the chosen method.	Leadership actively uses daily Kaizen to help build a habit of continuous improvement. Leadership rewards continuous improvement behavior. Leaders help to define new goals when existing goals have been met.	Leadership encourages organizational learning and rethinking of purpose through continuous improvement.

DASA MATURITY MODEL – KNOWLEDGE AREAS

	Level 1 – Novice	Level 2 – Competent	Level 3 – Proficient	Level 4 – Expert	Level 5 – Master
BUSINESS VALUE OPTIMIZATION	IT executes requests from the business without questioning their value.	Individuals learn the hard way that building unwanted solutions can be prevented by seeking contact with customers early in the lifecycle. Service levels are agreed for the service.	Initial feedback loops are put in place to ensure that the team understands how the users experience the service. Team bases its plans for the future of the service on feedback from users. Team monitors the behavior of the service and uses the information to steer improvements.	Team actively seeks fast feedback and involves stakeholders in the development and health of the service. Team creates plans together with user community.	Team is acutely aware of the business value of each adjustment to the service, and steers to optimize the delivery of value in relation to the 'health' of the service. Service Level Agreements are replaced by cooperative action and understanding between team and users.
BUSINESS ANALYSIS	Business analysis is solely done within the business. IT expects requirements from the business. Requirements are solely functional. IT fills the backlog based on their intepretation of what customers want.	Individuals within the team develop business analysis skills as a result of the interaction with customers and their growing desire to provide more value to customers. Customer fills the backlog; IT manages it.	Business analysis is a joint activity involving team and customers. This involves embedding business analysis as a skill within the team. Team actively seeks knowledge regarding business analysis. Non-functional requirements are added to the backlog.	Team develops a standard for conducting business analysis together with the customer. Joint decision-making regarding all aspects of the service (functional and non-functional). Building the backlog is a cooperative activity.	Business analysis is a joint responsibility ensuring that the capacity within the team is used optimally to provide value.
ARCHITECTURE AND DESIGN	Architecture is a largely isolated staff department that decrees the rules for creating and designing products and services. Mostly these rules are not followed.	Architects get involved with the teams actually creating and supporting the services. They take a more cooperative attitude and seek to understand the difficulties faced by the teams.	Architecture becomes a role fulfilled within the team. There remains a skeleton central architecture staff. The team takes responsibility for the application of architecture guidelines and rules. More attention for the quality of design.	Focus on quality at the source through rigorous use of agreed architecture styles and patterns. Detailed understanding of existing technical debt and application of structure and strategies to remove the technical debt.	Architecture is integrated into the team. Overall architecture is a group process and determined by the architects in the teams together. Architecture moves from being a discreet activity to being a continuous process.
PROGRAMMING	All programming is carried out exclusively by Application Developers. A single programming language or platform is mandated.	Operations engineers take responsibility to develop their own scripts and programs to support their work. Multiple languages and programming platforms start to appear within AppDev and Operations. There is no policy governing the choice or regulation of tools.	Team agrees on the tools that it will use initially to develop and support the service. Team realizes that it needs to invest in software engineering knowledge and skills over the full technological stack and lifecycle of the service.	Team seeks mastery in programming through understanding programming tools. Team is able to use multiple tools to achieve their goals. Team focuses on full-stack development and treats infrastructure as code.	All members of the team have programming skills. Engineering culture is embedded in the organization. There is a continuous search for mastery in programming.

DASA MATURITY MODEL – KNOWLEDGE AREAS

	Level 1 – Novice	Level 2 – Competent	Level 3 – Proficient	Level 4 – Expert	Level 5 – Master
CONTINUOUS DELIVERY	There is no Continuous Delivery knowledge. Components of the Development, Tesing, Acceptance, Prodcution (DTAP) pipeline are discreet and separated.	There is a central build server and version control system, and automated deployments for application binaries (code). Continuous delivery principles are applied ad hoc.	End-to-end build and packaging from a continuous integration server. Single builds for all environments. Continuous delivery principles are applied regularly.	There is a pipeline enabled for delivery validation, and end-to-end automated deployments including most infrastructure components.	Continuous delivery pipeline that facilitates end-to-end change delivery, from development to production. In addition, all team members work with a Done=Live mindset and aim for putting new code in production daily.
TEST SPECIFICATION	Tests are specified once programming is completed. Testing is only done by test professionals.	There are mainly static code analyses and automated unit tests.	Regular stakeholder demos are organised and automated feature tests are being used. A test-driven mentality starts to develop within the teams.	Test-driven and behaviour-driven development practices are leading within the teams. Teams regularly apply test automation, also for non-functional requirements.	Testing is seen as the core of creating new functionality. Team have adopted business-driven, feature-based advanced test capabilities and automated testing is common practice.
INFRASTRUCTURE ENGINEERING	Specialist infrastructure engineers are the only ones allowed to touch the infrastructure. Each environment is bespoke.	Operational management of infrastructure is an explicit topic for which a dedicated platform teams takes accountability. Periodic checks of availability and capacity are done.	Infrastructure is self-healing, all components are redundant, no manual changes are allowed for provisioning.	A self-service portal with graphical UI is in place, including monitoring of platform health and status. Immutable infrastructure patterns are applied to the next-gen infrastructure.	Infrastructure is fully software-defined and is based on standard templates that can be easily adjusted. A modern container-based platform including continuous delivery services is provided as self-service to all business system teams.
SECURITY, RISK AND COMPLIANCE	Carried out as separate functions that create control steps after one or more process steps causing delays. Often built in as an afterthought.	Each technical area ensures that it complies with the security and risk regulations as set out by the Security, Risk and Compliance 'office'.	Security, risk and compliance are seen as the responsibility of the team. Interaction between team members ensures that security and risk regulations are both defined and executed by the team. Guidance and coordination comes from a central compliance organization.	Security, risk and compliance are part of the architecture of the IT organization. Team members fulfil the role of security and risk officer. Each team provides input for and steering of the security and risk rules together with the CISO. CISO provides input from external regulators.	Fully integrated into the teams' way of producing and delivering products and services. Teams also scan the external world for risks and regulations.