DASA

DEVOPS AGILE
SKILLS ASSOCIATION

ROLE OF SPECIFICATION AND VERIFICATION IN A DEVOPS ENVIRONMENT

By Luca Ingianni and Falko Werner

Luca Ingianni

Independent DevOps Consultant

An aerospace engineer by training, Luca Ingianni has been working as a consulting engineer for over a decade. His experiences working alongside smart people in many teams have convinced him of the importance of DevOps as a way to create great products.

His main focus is on keeping the entire value chain in view: after all, a great product will emerge only if specification, development, QA, and operations work hand in hand.

His goal is to make everyone's life better: the customers, by making outstanding new things available to them, as well as the engineers, by making their work successful, stress-free, human, and just plain fun.

Specialities: Odd forms of DevOps. DevOps for Embedded Systems, IoT, industrial production, regulated industries, and just anything off the beaten path.

Falko Werner

Independent DevOps Consultant

As a computer scientist by training, Falko Werner has been working in product teams for over two decades. His experiences when working with Product Owners, developers, testers, Ops engineers, DevOps engineers, project managers, architects and other Scrum Masters in various organizations have proven DevOps to be a reliable way to create great products.

His focus is on keeping an eye on the people working to create the product: after all, a great product will emerge only if teams are motivated by a stable and secure environments where they can develop mastery in their craft and believe in a product vision to work towards.

This is a foundation for creating products customers love, use and buy while teams enjoy improving the product and their work environment day by day.

Specialities: DevOps with many teams, transformation processes, online and distributed systems in manufacturing, infrastructure, mobility and insurance enterprises as well as everything that has to do with energetic people.

DASA DEVOPS AGILE SKILLS ASSOCIATION

TABLE OF CONTENTS

CORE MESSAGE

In today's fast-paced business environment, organizations can't afford to drop the ball. It's crucial to ensure to build the right product, and ensure the right product got built.

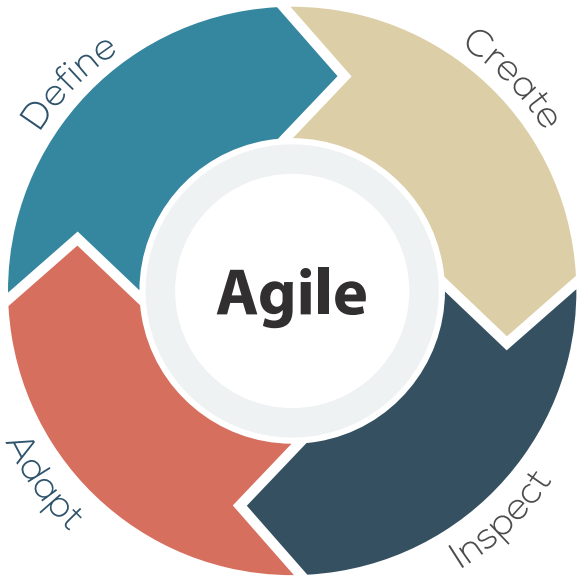
INTRODUCTION: CREATING GOOD CODE ISN'T ALL ABOUT CODING

It's reasonable to expect your code to be correct, well-written, maintainable and many other things.

But how often have you seen products that work flawlessly, beautifully –and solve no problem you would care about? Or, conversely, excellent ideas that fall flat on their face because the product is so brittle, so technically bad that it's unusable in practice, and all the brilliant ideas just wither and die in a sea of bugs.

It's a crucial issue to not just practice the art and craft of writing software well, but to ensure to build the right software, and to build the software right.

SPECIFICATION, VERIFICATION, AND THE AGILE WORLD



Specification! The word alone may send shivers down some readers' backs, awakening memories of 1000-page monstrosities lobbed at them by meanly snickering business analysts.

Aren't we now in a more modern, Agile time? Where such documents have gone the way of the dinosaurs? Of course we are. But if anything, the importance of specification and verification has grown, even if their shape has changed.

Product creation has sped up considerably. This means you can't afford to create something wrong, to notice too late that what you made isn't working. The only chance you have is to speed up along (and, hopefully, ahead of) the competition.

And this means tight, fast, powerful feedback loops, stretching from the intentions of the customer, through development and operations to the product in front of the customer. At its core, this is what specification and verification are all about.

This DASA profile encompasses the fundamental understanding, the mental and technical tools to reach such speeds – sustainably.

PRODUCT DEVELOPMENT IS A MARATHON, NOT A SPRINT

If you want to create great products, you need the entire development process to work well. If you exhaust yourself or falter somewhere along the way, you won't reach the finish line before they take down the garlands, and perhaps not at all.

Many teams have a focus on the actual development process. They are missing an important point: what comes before and after writing code is just as important. And just as hard, if not harder!

And, equally important: not only do all parts need to be executed well: they also need to work together, to create fast, powerful, uninhibited flow of ideas into working products; and provide feedback into the development process, to generate even more ideas and hone their application.

So, let's talk about what happens "around" software development: before code is being written, and after.

SPECIFICATION: WITHOUT CLARITY, YOU ARE LOST

First, you need to have a clear sense of direction.

If you don't get your specifications right, your team will be at sea in a fog.

They have no other choice but to slowly, carefully guess their way forward.

I'm sure you can imagine the results: even if you don't run aground, you'll certainly bump a reef here or there, completely unsuspecting.

Your sailors will lose confidence and trust, just kinda hanging on for dear life – or jump ship.

Clarity comes from continuously keeping your users' needs in focus and validating if what you're building is fulfilling those needs.

There are many frameworks and tools to help you do this. So now picking one has become a challenge all of its own.

Once you've decided on one (or a combination), there's still the matter of actually applying it correctly.

And you'll need to find your own way: because specifications are nothing but communication – between humans. And humans are very different, and so are their situations. You'll need to find a way to tailor these tools and frameworks to the humans you're working with. This is the real secret to specification: getting the human aspect right, getting everybody to share, transport, refine ideas, and concepts. Once that's done, you'll find the technical work to be the easy part.

On the other hand: if you can't achieve that, you'll never get that sense of direction, that common view and working hand in hand – and you'll be stuck tumbling through your product development, turning out something amazing only by coincidence.

Wouldn't it be great if you could create amazing products time and time again?

CREATING PRODUCT FLOW

How do you get into applying specification methods in a way that makes sense for you, for your team, for your situation?

DevOps is built upon the principles of Agile and Lean. Agile is built around working in short, meaningful iterations with maximum feedback. Lean focuses on flow or the continuous, unhindered motion of work through the development process which creates working software or hardware that can elicit the necessary feedback. And, communication is the wind that drives you forward.

Meaningful Specification

Feedback and flow demand, first of all, a solid foundation: good, meaningful specifications which clearly capture the customer's intent and path to value creation, as well as communicating them clearly to everyone. This includes visualizing them to make the insights explicit and actionable.

Systematic Feedback

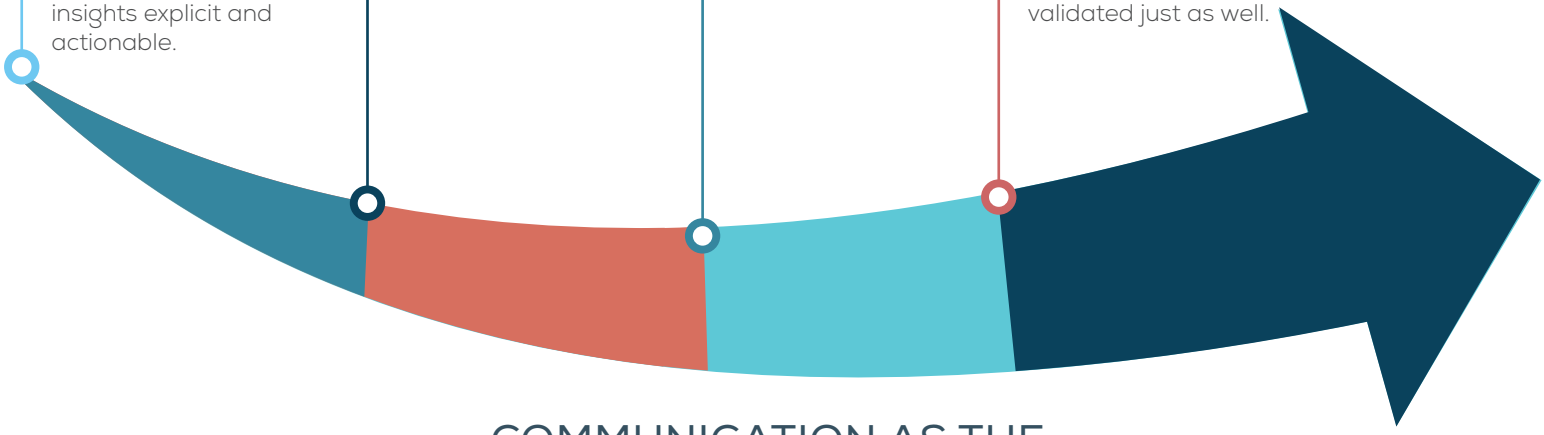
The second part is an equally powerful, fast, reliable system of feedback mechanisms. These include feedback within the team, tests of various shapes, as well as feedback from production, or straight from the customer.

Visualized Information

Yet again, visualization is a key technique to spread information throughout the team and the rest of the organization.

Well-Communicated Environment

Specification and verification do not only apply to the product itself but also to the environment they are embedded in. Architecture determines and enables the product that is being built and needs to be understood, communicated, and validated just as well.



COMMUNICATION AS THE INGREDIENT OF FLOW

UNDERSTANDING YOUR CUSTOMERS

Identifying the people who ultimately use the product or system you are building and understanding what they need in their context is key to calibrating the team's compass on the right course. Specification is not just a way to write down details that nobody reads but an active and iterative activity which clarifies what is the "right" thing to build. You can only get it right, if you approach the specification from the right perspective – those of your customers.

SPECIFICATION FROM THE RIGHT PERSPECTIVE: UNDERSTANDING YOUR CUSTOMERS

DevOps borrows its product focus from the Agile way of thinking about the world. What product should we build? Why? For whom? And... which product should we not build?

ARCHITECTURE'S ROLE IN SPECIFICATION AND VERIFICATION: LOOK AT THE FOREST, NOT THE TREES

Here's a dirty secret for you: specification, verification, and architecture mutually influence one another, in fact are unavoidably linked.

Along which interfaces is the entire system split up? How do the teams work together? How does a single team work together, internally? Which information is needed and how and when?

The art is in creating an architecture that works well for all ingredients of the system: the technological parts as well as, yes, the humans involved.

Architecture isn't just about APIs and interfaces and which server sits where. Quite crucially, it shapes how we think and communicate about what we are building: it provides the nouns we string together as we discuss our product's inner workings.

And conversely, how we think and talk about our product, how we organize our teams, unavoidably shapes the architecture which emerges.

VERIFICATION IN DEVOPS

Verification is sooo '80s! Who has even time for that anymore? We need to ship, and quickly!

...or is that true?

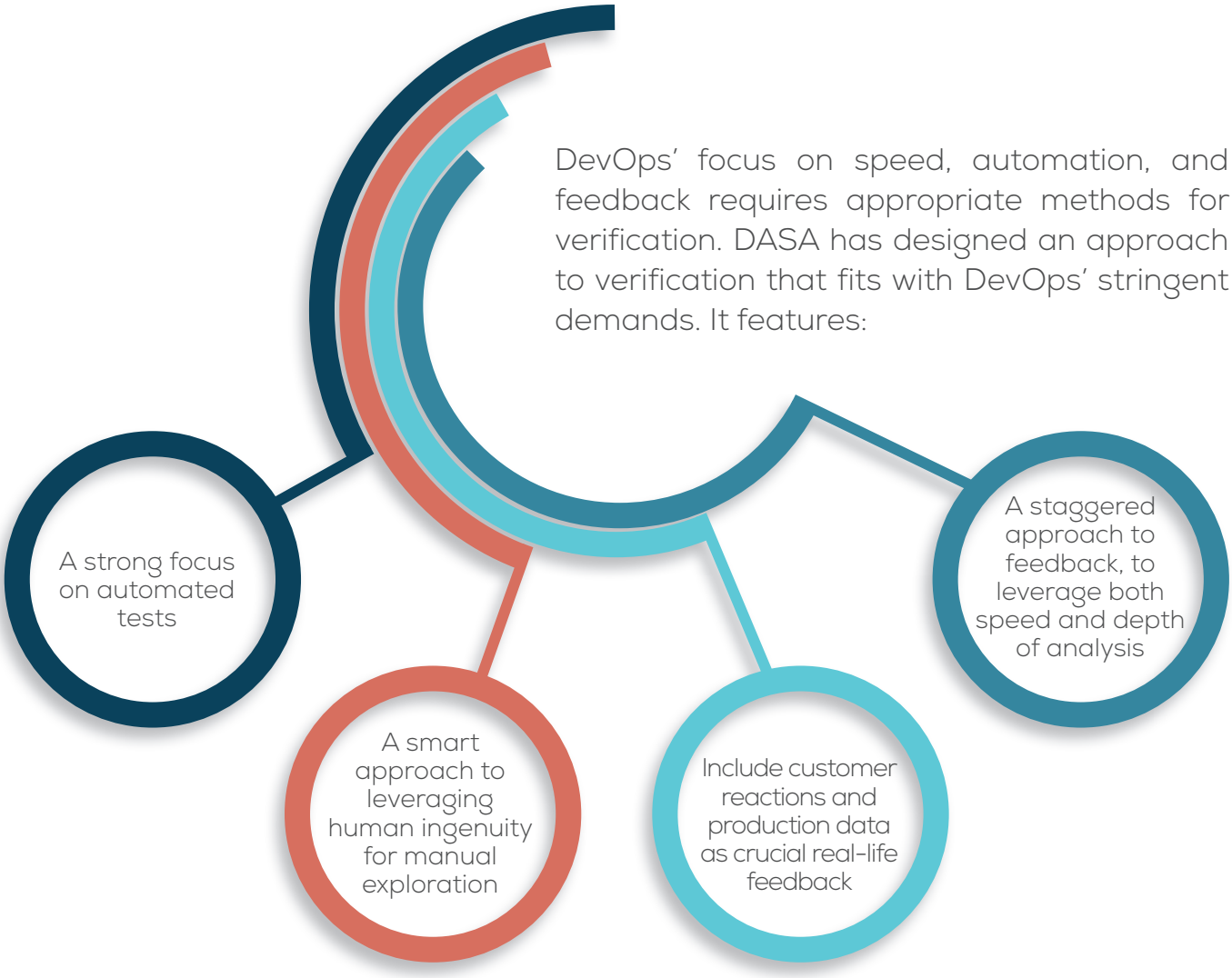
Of course, it isn't.

Verification is probably more important than ever: the Agile revolution that has taken place in software development hinges on quick, accurate, comprehensive feedback.

We've already established the importance of specification. Its necessary complement is verification: ensuring the goals laid out in the specification have actually been reached.

However, one thing is sure: along with 80s hair styles, 80s verification methods have gone out of style.

DevOps' focus on speed, automation, and feedback requires appropriate methods for verification. DASA has designed an approach to verification that fits with DevOps' stringent demands. It features:



VERIFICATION CHALLENGES IN A DEVOPS ENVIRONMENT

Before a product can be released, it needs to be found to be of adequate quality.

Well, fine, agreed. But what do these three words mean: "found", "adequate", and "quality"?

Before we can determine the quality of something adequately, we need to understand what quality is. What does it mean for our product to be of high quality? Does everyone have the same understanding of this meaning? So the first step in any QA effort will be to find answers to those questions.

The second, similarly conceptual step will be to understand the demands on our product. Are we building a kid's toy? A cardiac pacemaker? (While most people will consider a pacemaker's flawless functioning to be of higher importance than those of a toy, parents of toddlers may beg to differ).

Only after an understanding has been reached of the degree of trust, we require to place in our product, can we set out to ensure that whatever we created is in fact trustworthy enough.

Traditionally, much of that work was achieved through testing. But in the new, fast-paced, highly integrated world of DevOps... how will that work? What does verification look like in such an environment?

In fact, what does that environment look like in the first place?

AUTOMATION

DevOps has a strong focus on automation. This has profound effects on how tests are designed, when they are run, and when and how the test results are used.

CI/CD

In particular, the immediacy of using continuous integration to execute tests suites automatically the instant new code is created by the developers requires changes to the way organizations work together – and in turn offers new opportunities to change our way of working for the better.

MODERN TESTING APPROACHES (TDD, ATDD, ...)

Hold on, some of you may complain. This isn't new anymore. In fact, it's decades old.

You would be right. But nonetheless, they are frequently not the norm in many organizations. And even if they are... DevOps has extended the idea of product. It transcends mere code, and focuses on product, on customer value. This means that the scope of things to test has expanded dramatically. Infrastructure can be developed in a TDD style.

The product developed by a team extends past mere lines of code, to actual business value. Which will need to be tested efficiently, quickly, and thoroughly.

THE CRAFT OF TESTING, AND THE ROLE OF THE TESTER

The aforementioned aspects: Automation, Continuous Integration, and TDD change the approach to testing. Some go so far as to say that the role of tester is dead and gone.

What's certainly not viable anymore is the classic approach to testing: collecting a bunch of new features until the bitter end, then dump it all on some testers to make sense of.

Instead, if there is a dedicated test expert on the team, their role will move from execution to more advisory or architecture-related activities: creating the necessary fast feedback pipelines, ensuring that the emerging architecture fits the pipelines (and the other way round), helping the rest of the team move towards efficient verification as quickly and thoroughly as possible. This may also mean eschewing pre-release tests in favour of verification in production.

VERIFICATION IN PRODUCTION

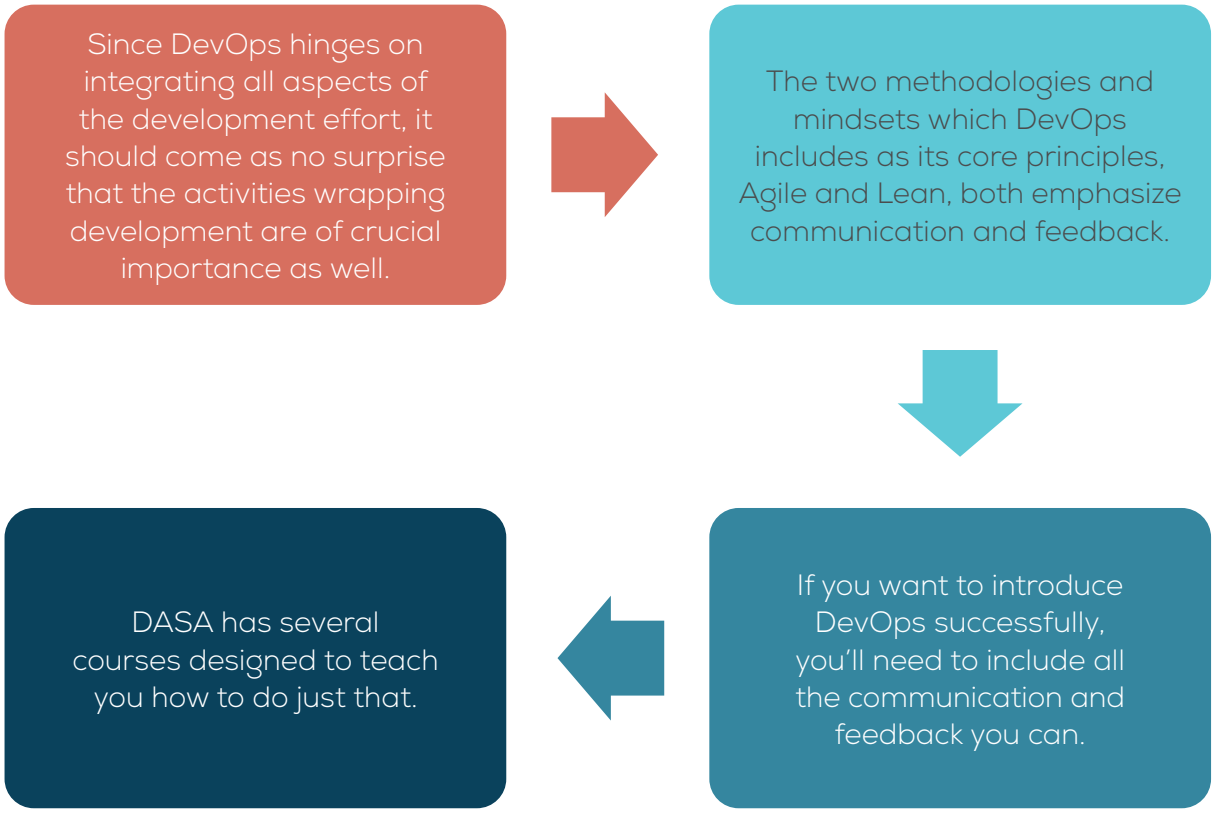
Frequent, bold releases to production require careful attention to how the product is behaving in production, how it's received by the customer.

Of course we'll interact directly with the customer to get their voice, but stringent automation and clear communication all through the development effort give us new opportunities to create better products in less time.

Going into production quickly and frequently (in the case of Continuous Deployment, even immediately) may feel risky and threatening – yet the opposite case can be made. If done right, risks will be lower, while we reap new benefits that were impossible before: new insights into our customers' wishes, lightening-fast reactions to new discoveries, upcoming events or threats.

But moving this fast without breaking things means having a reliable, solid, well-thought-out monitoring system. And that very much includes fast, powerful feedback loops which enable us to immediately detect and react to new events, be they beneficial or threatening. Such systems include technology, but more crucially they encompass the human aspect: having engineers' brains in the loop, and having a team and indeed company culture that is equipped to deal with the degree of transparency, speed and trust that a DevOps organization both provides and demands.

CLOSING THE CIRCLE: THE ROLE OF SPECIFICATION AND VERIFICATION IN A DEVOPS ENVIRONMENT



APPENDIX:
DASA COMPETENCE MODEL

The DevOps Agile Skill Association (DASA) Competence Model identifies 8 knowledge areas and 4 skills areas that are relevant in DevOps. Every individual operating in a DevOps team require to be competent at all 8 knowledge areas and proficient at the 4 skill levels. In a search for ultimate flow, at the team level, one should strive for level 4 for each competence level.

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

DASA Competence Model

DASA DEVOPS FUNDAMENTALS

This certification provides an extensive introduction to the core agile DevOps principles covering the essential knowledge and skill competences that have been defined by DASA. It is the starting point for an organization going on a DevOps journey. Improved workflows and faster deployment starts with a core understanding of DevOps fundamental concepts by anyone involved in an agile and/or DevOps team.

DASA DevOps Fundamentals brings all Knowledge and Skill Areas to the Competent level. Having said this, the Fundamentals course does not provide, for example, practical Business Analysis or Programming experience. Such practical knowledge needs to be acquired through specific training.

DASA DEVOPS PROFESSIONAL: ENABLE AND SCALE

This certification brings the Skill Areas to the Proficient level. Enable and Scale is about ensuring that a DevOps team can be successful in its environment. The Enable and Scale certification:

- Validates the candidate has a practical understanding and experience in leading DevOps teams.
- Enables members to become effective ensuring the team works optimally.
- Covers the effective coexistence and cooperation of multiple DevOps teams.

In other words, it answers the question "How do you turn a traditional IT organization into a DevOps organization?"

DASA DEVOPS PROFESSIONAL: SPECIFY AND VERIFY

This certification focuses on competencies: Architecture and Design, Business Value Optimization, Business Analysis, and Test Specification.

According to estimates, approximately a third of the team is involved in these areas of expertise. The focus of these areas is to ensure customers' requirements are understood and translated into a team. In other words, these knowledge areas confirm that the requirements can be integrated into an IT service.

The ultimate responsibility of the role, Specify and Verify, is to ensure the design of the service is future proof, both technologically and functionally. In addition, the role confirms that the ability to test any new functionality is optimally facilitated by Test Specification that takes both the customer usage of the system and the need for speed into account.

DASA DEVOPS PROFESSIONAL: CREATE AND DELIVER

The majority of a DevOps team falls into the competency area of Create and Deliver. The core knowledge areas of this competency are "Programming," "Continuous Delivery," "Security, Risk, and Compliance," and "Infrastructure Engineering."

This competency is really the heart of the team's capabilities. The exact balance of skills, particularly Programming vs. Infrastructure Engineering, depends on the total area of responsibility of the team. DevOps is being used to deliver

infrastructure services, which in turn deliver application services. A vital aspect that helps deliver a successful service is monitoring. In practice, full stack DevOps teams responsible for everything from user interface to server and network hardware are rarely seen. It is the technology stack that ultimately determines the balance of knowledge required in a team.

DASA DEVOPS PRODUCT OWNER

In a DevOps environment, the Product Owner is a critical leadership role and responsible for managing the full lifecycle of a product from concept to the grave. This certification program helps the Product Owner realize maximum business value, engage with stakeholders, and deal with future requirements as well as operational challenges.

DASA DEVOPS LEADER

The DevOps Leader is responsible for leading the DevOps initiative and creating the framework for teams to scale and achieve maximum business value. This certification program helps leaders understand leadership in the context of DevOps, discusses leadership development models, building teams, and transforming the organization.

DASA DEVOPS COACH

The DevOps Coach helps team members and other stakeholders in the organization to apply DevOps concepts and principles within their organization. The coach oversees the transformation and guides the organization through their journey in building high-performing teams.

www.devopsagileskills.org

JOIN OUR GLOBAL MOVEMENT

Do you or your organization want to join other leading IT thought leaders from industry, software, training and consulting organizations working cooperatively to build the most open, comprehensive platform available in the market for DevOps and Agile Skills Development? If so, you can become a DASA Forerunner, Training or Courseware Partner.

CONTACT

3201 Edwards Mill Rd, Ste 141-605
Raleigh, North Carolina 27612
United States

www.devopsagileskills.org
info@devopsagileskills.org
+1 (302) 476-2732

/company/devops-skills-association

@dasa_org