

Build services faster

Make impactful ideas
take flight by accelerating
digital development

 FUJITSU



Building services faster will enable you to survive and surpass

In today's volatile and unpredictable business landscape, change is the only constant. Rapidly evolving customer behavior and demands driven by social, environmental, and economic pressures, are a major challenge for many organizations to keep up with.

In our recent *study*, over two-thirds of organizations said they needed to respond faster to changing market demands, and around three-quarters said they were prioritizing customer experience as the driving force of their future transformation strategy.

Time-to-market is therefore emerging as a top priority over cost, as well as continuous iteration to ensure that the product represents the aforementioned social, economic, and environmental changes around it.

The pursuit of better CX to win new hearts and minds – and maintain existing customer loyalty – is not a new challenge. However, it has taken a completely different course over the last year and has accelerated to a previously unimaginable speed.

Before COVID-19, consumer buying and engagement patterns had already dramatically changed due to digital disruption. The key trends were convenience, speed, or sometimes novelty – and the ability of modern applications to predict and serve these needs intelligently, and even proactively as new challenges continue to emerge.

However, more recently the change in attitudes and expectations has been due to necessity, as demonstrated by COVID-19 and the disruption it had on consumers' daily lives. The pandemic has brought the need for digital engagement, retention, and evolution to the forefront, dramatically increasing the importance of building services faster.

These shifts are here to stay as we enter a 'next normal', with the speed of updates and feature releases now a hugely important factor. Adopting an adaptive, evolving, and always-on engineering strategy will allow you to respond at pace, and pace will determine who thrives and who struggles to survive in today's competitive landscape.

Opening thoughts:

Organizations are realizing the value of building services faster



Jason Daniels - Digital, Data & Cloud Leader at Fujitsu

Time-to-market has always been important, but the COVID-19 pandemic meant that many organizations had no choice but to throw their efforts behind rapid digital service development, as physical routes to market were forced to close. Some of our customers needed help to do this on a reactive basis to continue trading, with Fujitsu squads deployed to co-develop new digital solutions. Others, with our support, were ahead of the game. Take one of our public sector customers, for example, who had already worked with us to digitalize their core services to make them more convenient and accessible to the public. This gave them an excellent baseline to build upon when the challenges of the pandemic arose.

Using a low-code development approach, together we were able to design and deploy a new application and deliver subsequent updates faster, to extend the reach and enhance the user experience of its 50+ physical locations. The new application platform enables the organization to engage a community of hundreds of thousands of people, both directly and via self-service, while meeting its industry and geographical regulatory needs.

Key initiatives for you to focus on

As demonstrated, many organizations are coming to the realization that building greater adaptability and pace into digital service development is vital. The top-performing ones are discovering the transformative power of aligning modern technology, processes and people behind this need for speed - with a fusion of Cloud, Agile and DevOps best-practice adopted to build services faster.

However, plenty of organizations are yet to transition from the traditional application 'change and release' methods that may have served their 'business as usual' for a long-time, but that are no longer fit for a world of unpredictable change and rapidly evolving needs.

We have identified three key initiatives for this, so you can achieve the high-velocity application lifecycle management needed for the future of your digital services.

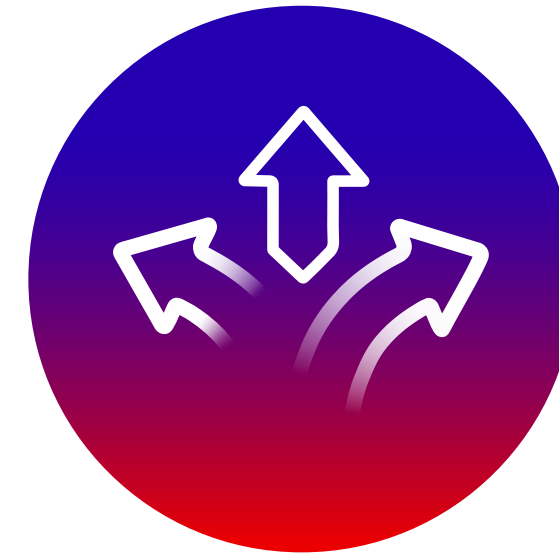
These are explored throughout this guide.

All organizations need to be able to get three things right to achieve high velocity application lifecycle management:



1. Architect for velocity

Moving from monolithic, towards decomposed



2. Agile innovation approach

The right team formation, culture, and control



3. Addressing resource gaps

Overcoming skills shortages with people and technology

Achieving this gives you the required speed across technology, processes, and people – and makes it possible for application development, consumption, and relevance (through regular updates) to be achieved faster.

1. Architect for full-stack velocity

Creating the optimal application deployment characteristics – namely architecture and underpinning technologies – will play a pivotal role in accelerating the development, deployment, and evolution of digital services.

Considering this, the ideal landscape is an IT service ecosystem that is able to responsively shape and prioritize fluctuating business demands - and deliver on these in the form of new solutions and features that can continuously evolve to ensure future value.



Productize for pace

More than two-thirds of IT leaders in our recent study said that applications were going to be key to enhancing customer experience while accepting they needed to respond faster to customer demands. However, while applications (both front and back-end) are an important component, organizations should in fact re-evaluate whether their whole stack – and how they deliver to market – is fit for a digital age that demands adaptability as standard from top to bottom.

Often, we see too much focus being placed on the front-end external application or digital service. However, this is not a sustainable solution because most releases and updates will have dependencies elsewhere in the stack. If these areas aren't proactively addressed then the pace of front-end development and time-to-market is constrained, or even completely undermined, due to burden and bottlenecks elsewhere.

Productized development is crucial for full velocity and adaptability - with the full-stack being treated as a unified and completely code-based entity, along with clear ownership and high-automation supporting its continuous evolution.

Expert view:

Embracing full stack development and everything as code



Jason Daniels - Digital, Data & Cloud Leader at Fujitsu

One of the big challenges I see in organizations is fragmented ownership. Full-stack development capability can help by creating an empowered single point of ownership, minimizing development constraints like complex communication channels, cultural disconnects, manual errors, and bottlenecks caused by slow supply of resources. I'm also seeing an evolution of how applications need to operate. Client-side processing is taking on more of the tasks which have traditionally been performed on the server-side, and a wider variety of databases and data abstraction methods are being used across the application stack. By acknowledging these changes, full-stack development enables you to scale at speed while using the right back-end data model to ensure quality service is always maintained. 'Everything as code' augments this by providing the standardization needed for faster and more automated execution, reduced capacity for human error, easier continuous iteration, and greater continuity as resources change over time. These benefits can be taken to the next level by applying machine intelligence to interrogate code even quicker and have it automatically checked throughout development to ensure a consistent high-quality in change release.

Decompose to deliver at speed

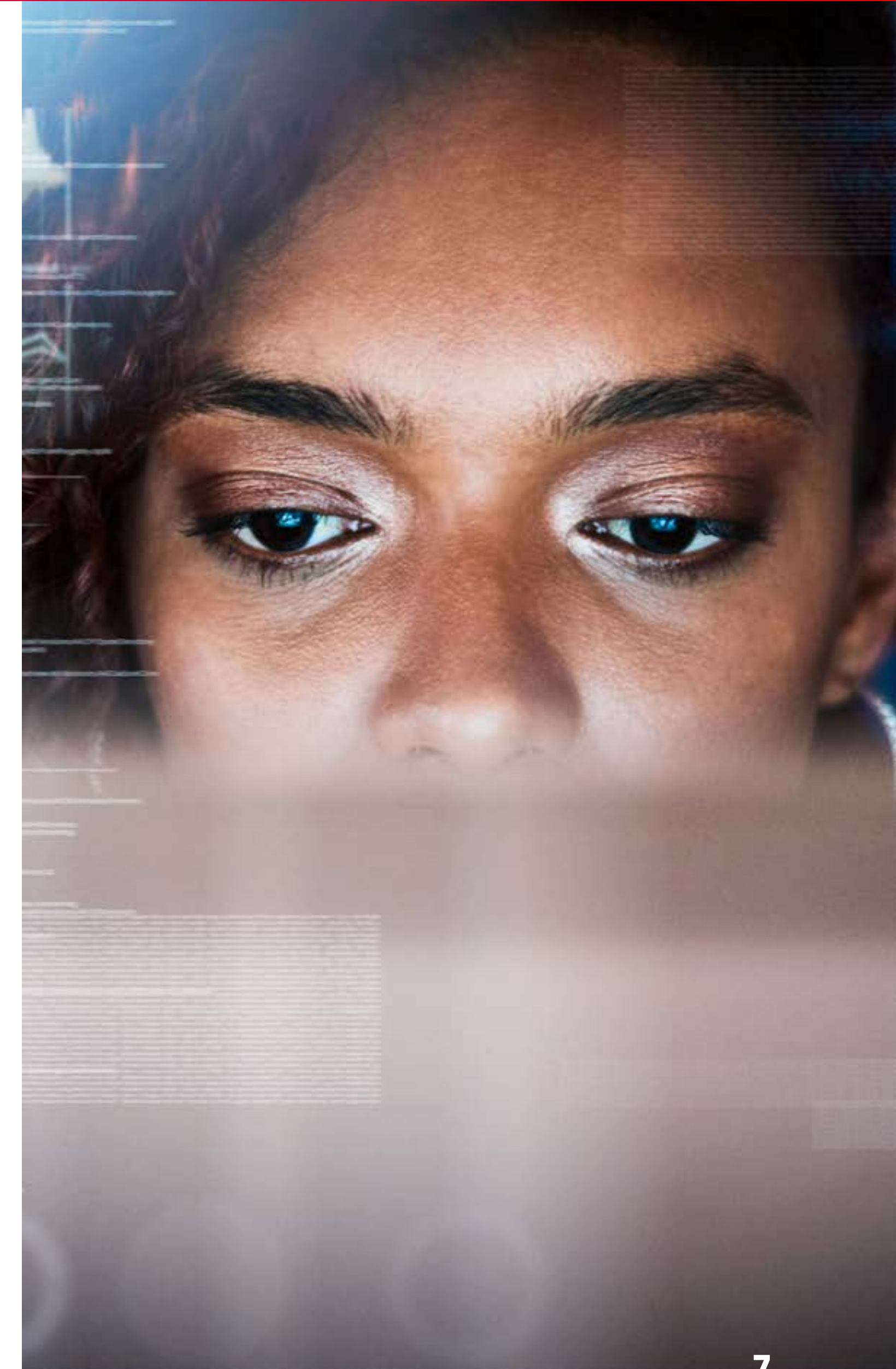
When it comes to applications specifically, you need the right architecture to achieve the pace and adaptability that your productized stack intends to deliver. For most organizations, their current legacy application architectures are not flexible enough to compete with the more digitally mature organizations – but why is this?

It's because of a lack of composability – and for this reason the biggest step required is for organizations to move away from a monolithic make-up, towards an agile and decomposed approach to development and delivery. This involves building much smaller units in the form of containerized microservices and even event-based code to enable speed, but more importantly, zero downtime as and when updates are made and new features are released.

The ways this works in practice – and one of the key concepts you need to adopt – is to treat applications as discrete business value

components connected together (via an API integration hub, for example). These can be exposed either as low-code modules to empower a citizen developer community, or programmatically to pro-code developers, within an organization. This helps to ensure business data and value can be consumed across the whole application landscape and not siloed into single areas.

The most successful and powerful examples of this are the disruptive digital platform businesses of recent years, who can alter their content, services, and aspects of their user experience at rapid speed. They are leading the way by operating the composable application estates of the future, and the result is that the features of their digital services can be quickly assembled, re-assembled, and adapted to meet omni-channel business demands and consumer trends.



Codify and connect

We've seen the composability trend starting to gather pace within more traditional organizations too, accelerated by the unpredictability and constant flux throughout the COVID-19 crisis.

These organizations either adopted modern, composable architectures to bring their services into the digital era, or in some cases went even further by using them to extend into other market channels. In best-practice cases, this presented an opportunity for expansion and growth - and, at worst, provided a means of offsetting the financial losses of closed physical outlets during lockdown.

Great examples here are retailers and fast-food outlets, who extended their usual operations via third-parties for rapid home delivery fulfillment at a time when everything had to be consumed remotely; or the restaurants and bars that introduced an 'order at table' functionality which enabled the safe return of public gatherings; a concept which has since become embedded as the norm within many organizations' service models.



While many of these use cases involved codification of existing non-digital assets, such as menus to enable third-party delivery of food, they also prove that creating a modern applications experience doesn't always require a costly or complex modernization programme.

Of course, becoming composable in these scenarios did involve building some new cloud-native functionality. However, arguably a more crucial task for delivering at pace was to extend the enterprise applications' data that underpinned the existing core service, as opposed to engaging in a broader modernization or re-build of legacy infrastructure and applications.

In such circumstances, particularly where short timeframes and/or the lack of a mass developer pool are proving to be barriers, APIs and low-code solutions are often crucial for lowering these; making it possible to deliver powerful omni-channel services quickly and cost-effectively.

At the same time, they provide a highly available legacy integration route to ensure a singular data plane. This is typically of huge value to future development projects, as both the applications landscape and its channels of consumption become even more diverse and distributed.

Expert view:

Re-build and re-use... combining forces for rapid digitalization



Jason Daniels - Digital, Data & Cloud Leader at Fujitsu

The last couple of years have been a major turning point for how organizations think about their routes to market and the experiences they deliver. Since the pandemic started, the number of products and services we have all consumed remotely or at a distance has skyrocketed – and I've often wondered, for all the (usually) seamless transition on the outside, how much chaos was going on internally within organizations as they used digital development as their primary vehicle for pivoting to new ways of sustaining and adding value? If we take the example of food and drink outlets, ordering my meal from my table, car, or home might have been an easy experience, but as I indulged, I often thought about how much complexity organizations grappled with to make this possible - particularly with very little time to adjust and to do so without an excessive cost or resource outlay. The solutions created, many of which involved codifying and connecting menus for partner distribution, highlighted that a fantastic, composable digital service doesn't always have to involve an end-to-end build (or rebuild) of applications. The rise to prominence of low-code and API platforms being maximized has meant that these concepts will surely now be exploited by many more organizations, especially those looking to quickly unlock future growth opportunities.

2. Agile innovation approach

Despite the potential of architectural transformation, this alone cannot deliver the target outcomes of high-speed, agility, and improved CX. There's something just as powerful that makes all of these things come to life: the shift in organizational approach and culture.

Many organizations have seen their technological advances rendered meaningless due to an inability to form the teams, master the methods and normalize the ways of working that facilitate business-wide agility and speed. Those that have made good progress are not only building services faster today; they are building an inclusive developer community culture which enables sustainable success in future.



Team formation and culture

Firstly, forming the right teams is crucial. Rigid structures and siloed resources are not fit for developing and supporting business needs at the pace now required.

Based on our experience with customers, we believe that small, multi-disciplined development squads that are assembled and embedded within business functions are the way to succeed. Furthermore, these business functions should now be less formal and hierarchical; rather, they should be more project-based, around a particular business or customer outcome.

To enable these teams to build services faster, organizations should adopt an autonomous approach, involving non-centralized decision making from the core, with devolved authority given to development teams on the ground.

With people empowered to understand the business requirements, they can make decisions and take actions on how to deliver continuous value throughout the new application management lifecycle.

In terms of informal leadership, the 'outside-in' approach based on customer requirements demands a product manager who is obsessed with customer experience. They should be capable and enabled to understand the business challenges, and through close integration with the teams, engage in sprint cycles that aim for a minimum viable product to enhance, rather than a perfect solution to the problem or target outcomes all at once. With this outlook, teams should be able to work iteratively through challenges and pivot in various directions based on external change.



From a cultural perspective, the adoption of continuous integration and continuous delivery/deployment (CI-CD) promotes an iterative developer culture to application development, empowered to make or automate small improvements on a constant basis, ensuring each development cycle is aligned with the business demands that are regularly fed through.

This mindset goes hand-in-hand with the composable architectures (de-coupled and API-led microservice applications) and event-driven code execution explored earlier. It allows teams to build services faster through rapid construction and deconstruction of components.

Working in this way, they are empowered to add, adjust and remove features based on rapid testing and feedback - and ensure that once new releases are pushed out for consumption, all aspects are highly available, resilient and scalable to meet future needs and growth in demand.

Expert view: Empowerment needed for continuous evolution



Jason Daniels - Digital, Data & Cloud Leader at Fujitsu

Culture is often one of the most overlooked aspects of accelerating digital service development. With a mindset of IT being embedded in the business, bringing together a community of the best brains for a particular project is the way to succeed. As seen within the many customer environments that our development squads have supported, multi-disciplined teams that are focused on a distinct business challenge – and empowered to make change and continuously evolve – are key to delivering at a higher velocity than ever before. This is crucial for the consumers and societies which our clients serve.

The adoption of CI-CD in particular is helping to bridge the gap between the developers and business experts that should jointly create solutions. It enables agile, actionable feedback loops and, with the help of automation, accelerates regular and iterative responses to these in a way that enforces code quality, reliability and security each time. And with the emergence of low-code also providing greater language equivalence across teams, it enables a more diverse group of people (with different skill-sets) to get 'hands-on' with iterative development and contribute to continuous improvement together, in real-time.

Pace and precision:

Mastering DevOps

With speed to market now so crucial, DevOps has taken even greater significance as a facilitator of optimizing delivery - specifically minimizing risk and lead-time, ensuring reliability and delivering on the 'pace potential' of decomposed 'everything as code' architectures.

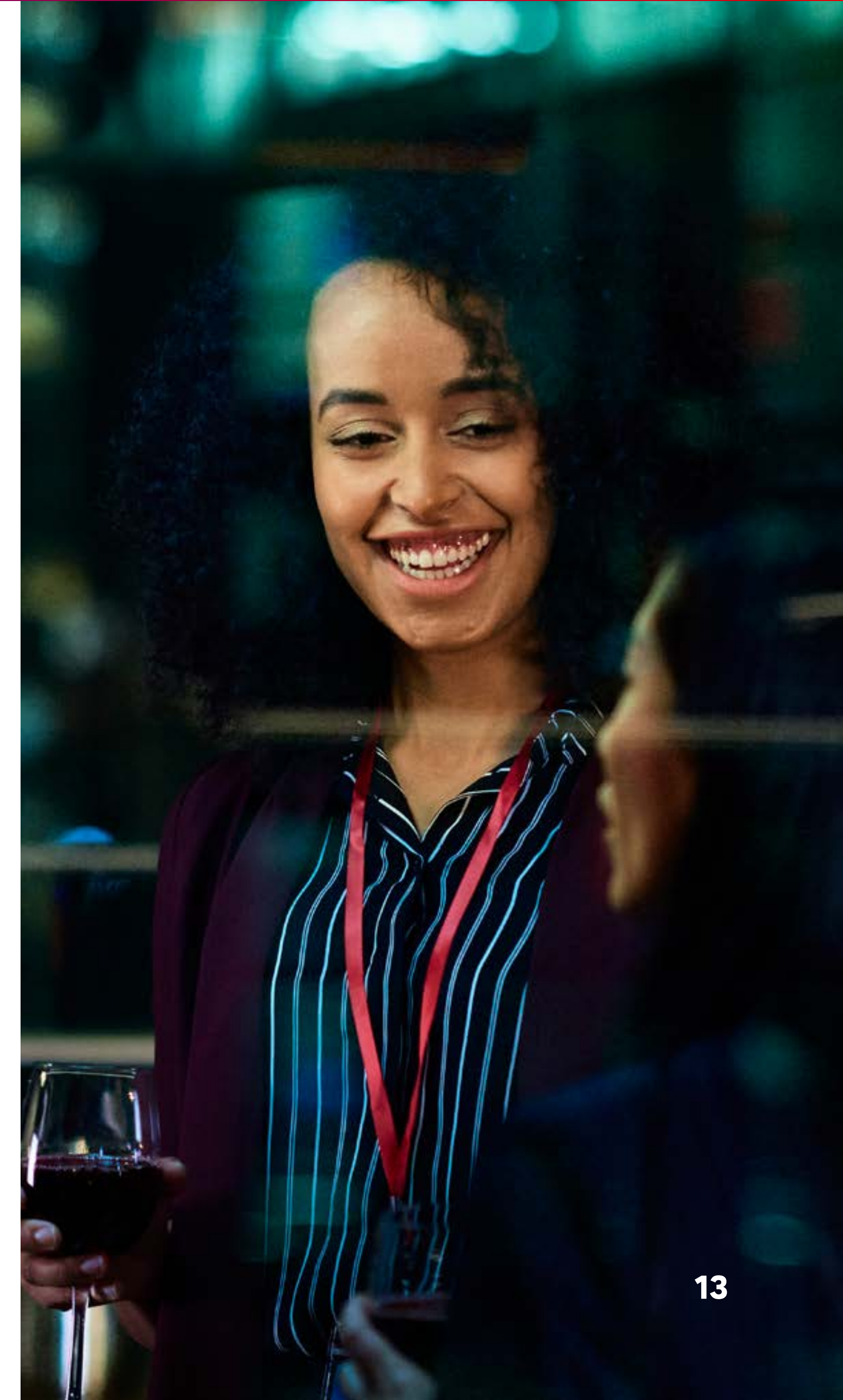
Yet many organizations are still to master the concept. There are many reasons why, but three significant mistakes we see are:

1. A technology focus outweighing a people and culture focus.
2. Deploying DevOps with traditional commercials.
3. Addressing both of the above but still abiding by outdated change/service management processes.

Firstly, there have been significant technological advancements in support of DevOps. In particular, automation and AI is enabling organizations to work faster, minimizing development inaccuracies, maximizing resource efficiency, and enabling more sustainable development.

However, as an inherently technical initiative, we still see many attempts at driving DevOps practices from a 'technology & tooling first' perspective, rather than a culture-led, change-initiative transformation.

To succeed, you must face the potential discomfort and politics of breaking traditional structures to connect two separate, multi-faceted worlds - Development & Operations - together as a single unit, with the same culture, way of thinking, and level of accountability throughout the application life cycle.



It is important that your people are empowered to learn from each other, creating an 'economy of excitement' which promotes skills transfer across the team, regardless of business background or existing skill-set.

Secondly, from a commercial perspective, we are still seeing organizations embarking on DevOps projects with set contractual and cost models based on traditional ways of working. As these do not allow for flexibility in terms of regular scope change or speed of release, they can often act as a set of brakes on the potential to develop faster.

The third constraint is where the focus and commercial model are correct but, where modern development and service management processes are required, traditional ones remain in place and act as a major constraint.

Implementing 'Agile', however difficult, needs to be prioritized. Converting long-established, monolithic processes into smaller, frequent, automated parts allows for fast and continual revision, particularly within 'change and release'.

Finally, if these three issues are overcome, then successful DevOps implementations should be scaled using a Center of Excellence model that contains repeatable development patterns, methods, and cultural norms for the wider organization to adopt.

Fast, flexible, and agile: Optimizing continuous development with the Andalusia Government

It is common for organizations to have DevOps-like initiatives or practices in place today, yet many still struggle to master the key concepts to deliver real impact (due to the constraints we have explored in this guide), or alternatively, have implemented small DevOps practices successfully but are finding it difficult to scale these for maximum value.

The Government of Andalusia got their approach right by working with Fujitsu. They partnered with us to accelerate the release and continuous improvement of crucial aid management services through a perfect fusion of DevOps, Agile, and Cloud, with expert engineering skills on-demand.

Working together with a Center of Excellence model, enabled by Fujitsu's Software Factory, we deliver full-stack, multi-disciplined development, underpinned by agile procedures and process automation tooling. The new way of working is fast, integrated, and flexible to changing needs - and it enables continuous development and rapid scale, thanks to the ability to re-use repeatable components.

Cloud-native autonomy vs cloud-native control

The creation, understanding, and use of cloud-native applications has now placed them front and center of many modern application strategies. It is no longer a concept adopted by the few for a bright digital future, but the very foundation of a relevant and competitive existence today.

However, while 'born in the cloud' development allows you to bring a richer set of features, functionality, and flexibility to the digital services you develop, the democratization and pace of development can result in unwanted business risk.

So, when it comes to using cloud-native as a means of accelerating time-to-market across multiple channels, it's important to put organizational and technical guardrails in place which bring a level of safety. From a platform perspective, this is best done in a tailored way

for each cloud, but with a lightweight, baseline framework in place to provide protection across the whole ecosystem.

To provide this to our customers, we deliver through our **Fusion Engineering** digital development practice. This introduces recognized and trusted agile methodologies, frameworks, and technologies to ensure we prioritize, develop, and deploy efficiently to the highest standards.

Fusion Engineering encompasses industry standard models such as SAFe, and is assisted by Fujitsu SpringBoard, which provides automated Infrastructure as Code templates to lower development risk, as well as the initial cost and lead-time for on-boarding to major cloud platforms including Amazon Web Services (AWS) and Microsoft Azure.



Accelerating with agile:

How Coop Trading reduced time-to-market by 300%

Facilitating faster development relies on the right approach and culture. It's equally important for introducing this speed safely, with minimal business risk. Many Fujitsu customers are mastering this through co-creation with our expert development squads. Take Coop Trading for example, which develops products for approximately 13 million consumers who shop across 4,000 stores. Working together with Fujitsu, they implemented Scrum as the agile delivery method, with two-week sprints and commitment-driven sprint planning.

We then built the foundations for a continuous delivery platform, through an automation tool-stack that supports agile development processes. The new way of working allows Coop Trading to get products to market 300% faster, and what's more, there has been zero churn in the service delivery unit for two years due to happy developers. In the words of Coop's Senior Project Manager "people want to be a part of the whole development process because we are making awesome solutions for our users now."



3. Addressing resource gaps

Quicker time-to-market requires digital development skills that are in short supply. Many organizations simply do not have the knowledge and experience to build services faster, and gaining and scaling this expertise in-house can be complex and difficult. This can be because of the range of competencies now required across a diverse range of languages and platforms, and because sourcing developers is now a highly competitive market due to supply vs. demand.

This second issue has intensified even further in the post-COVID era. Although the geographical scope for skilled resource has broadened due to remote working and the proven success of 'remote team development' projects, the competition for skills has also increased in line with this.



Enabling citizen developers

In future, organizations will need to 'open up' the opportunity and capacity to innovate to business-line stakeholders, rather than them simply being a point of feedback for the technically skilled.

Due to skills constraints, creating and then enabling this 'citizen developer' community internally will be key to scaling resource in the future. Today, many organizations are programmed to think of application development as a specific 'place' and set of people within the business, tasked with creating a user experience, ultimately to collect and manipulate data. But what if everyone in a business was empowered, from a structural, hierarchical, and technical standpoint, to do this whenever needed?

From our work with customers, we can see that enabling this has huge benefits for the pace and UX quality of digital service development.

This is because understanding the latest requirements, and then driving ideation, experimentation, and continuous improvement, is accelerated as a synchronized 'IT and business' set of activities within the small, autonomous teams mentioned earlier.

One of the leading methods of technical enablement is currently the deployment of, integration across, and training in, low-code/no-code practices and platforms.

The rise of these - offering a more visual, intuitive, and accessible approach to development (compared to pro-code) - is enabling organizations to simultaneously accelerate the release of applications and bridge crucial skills gaps.



Gartner research predicts that by 2024, around 65% of application development will be executed through low-code platforms, as organizations look to reduce the dependency on pro-code developers by abstracting application programming behind graphical tools and offering a library of ready-made resources for use by the less specialized.

Not only does this enable faster build times for smaller feature releases, but it also enables organizations to release valuable engineering resource to the bigger, more complex aspects of projects, where their skills are really needed. This reduces burn-out and overstretching of talent in the developer pool, which in turn makes 'building services faster' more sustainable in the long term.

While low-code technologies are powerful tools, another excellent long-term initiative which the most progressive organizations are driving, is to enable their typically 'non-technical' employees to build a range of engineering competencies as part of their professional development. This recognizes that in the modern era, every organization increasingly needs to become a software development house, and that by building digital literacy across the business, their core development community can be assisted to an even greater degree.

Life-saving speed:

How Rock2Recovery accelerated vital services with low-code

Low-code is far from a new concept but we are really seeing it scale to new heights in supporting organizations to build services faster.

Many of our customers are taking advantage of low-code development platforms such as OutSystems, particularly as its vast library of pre-built modules can be deployed and tailored with very little coding expertise.

Organizations like Rock2Recovery, a military charity who needed a rapid modern application to upgrade its critical and potentially life-saving support to veterans, used the platform and our expertise to develop a full solution in just days - achieved with very little outlay and a minimal developer pool.



There are some great examples of these digital proficiency initiatives, such as **BAT's Digital DNA program**, which encourages upskilling and a 'digital-first' mindset across business units. In such a program, innovation is made accessible to all employees, enabling them to understand and experiment with cloud-based development tools and emerging technologies in a safe environment. This way, they can build their capabilities and, as they gain competence, put them to use alongside the expertise of experienced developers in the real-world; assisted by safe guardrails in the form of blueprints, templates and low-code platforms which restrict the potential for human error.

But going forward, will humans be assisting humans for much longer? Or will machines provide the extra support when there are skills and resource gaps? As artificial intelligence (AI) continues to mature, it is likely that we will see a rise in the creation and adoption of 'AI-assisted development' technologies.

These will support pro-code developers to build services faster, by auto-coding based on predicted requirements and remediating human error in lines of code before it is pushed into the pipeline or production environment. With the proliferation of hyperscale platform solutions in this space such as AWS Code Guru and Microsoft Visual Studio IntelliCode, we predict this concept will only scale from here.

The power of co-creation

Trying to amass all of the required technologies and expertise in-house, without any external support, is a valid long-term ambition but most organizations have a much more immediate skills challenge to address. For this reason, we recommend a more co-creative approach that combines internal capabilities and the expertise of best-in-class partners, whereby technology-provider engineers work collaboratively as an extension of the in-house team.

This target, accelerated by the need for the growing customer demand to 'show, not tell' throughout the various stages of service, product, or feature development, is why continuity and flexibility of people is key. As well as helping to overcome skills shortages through combined depth and breadth of skills, knowledge, and experience, this model enables the in-house development team to upskill and

mature their own capability and knowledge-base through learning and understanding best-practice from outside of their usual comfort zone.

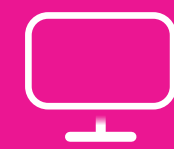
Because of this, we encourage all teams to have engineers with core competencies across a consistent set of nine areas; but with specialisms in some and an appetite to learn across others to become more well-rounded for the longer-term.

Scroll down to find out what the nine areas of core development competency are.



Nine areas:

Of core development competency



Applications

Competency in modernizing a range of application types, including when and how to migrate, modernize, and build new.



Data

Understanding of how to identify and exploit data safely and ethically through new applications, digital services, and API platforms.



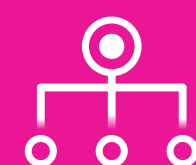
Platforms

Practical knowledge of development platforms and understanding of their features, benefits, and drawbacks for various scenarios.



Integration

Expertise to integrate distributed components including platforms, applications, and databases – from legacy to cloud-native.



Content & Experience

Skills in UX development and testing, as well as implementing agile feedback and CI-CD chains for rapid new releases.



Automation

Experience in using a variety of automation platforms and toolsets to accelerate delivery, with greater reliability and fewer defects.



Security

Full-stack security and DevSecOps proficiency including the measures, protocols, and configurations needed to proactively protect data.



Quality Assurance

Aptitude in test, release, and continuous improvement methodologies to ensure a high-quality experience for all end-users.



Reliability

Ability to build and enhance critical applications without disruption to live service, as well as deployment, monitoring, and failover.



Closing thoughts:

Should you tackle skills shortages through people or tech?



Jason Daniels - Digital, Data & Cloud Leader at Fujitsu

Despite incredible advances in low-code and even no-code platforms in the last few years, the challenge of skills is not going away. The need to have scale in 'on-demand' pro-code (especially cloud-native) expertise – crucially with the enterprise rigor and accreditation needed for relevant platforms and sectors – is only going to increase from here. There are some amazing 'AI-assist' technologies I've experimented with, and they really do enable developers to accelerate builds with the confidence that they aren't prioritizing pace over precision.

Overall, though, there simply is no substitute for skilled people – and that's why developing in a partner ecosystem is the only way for organizations to succeed. At Fujitsu, we know that resource augmentation can be needed at any point in the lifecycle – which means that flexibility and dynamic right-sizing of supply is crucial, but that continuity and familiarity of the people supplied is also important. That's why we support our customers in four modes – supply of individuals, small squads, large project teams to scale, and engineering consultants to define strategy. This 'dev on tap' model is delivering great results for organizations across the world.

Five key actions for building services faster now:

Accelerating development is complex and you may be looking for a place to start or some key areas to focus on.

Fujitsu provides a fusion of technology, people, and process expertise to help you rapidly create adaptable, product-centric services. We engage at any stage of the product lifecycle with proven agile methods to hit your objectives.

1

Ensure that your underpinning technology platforms can run and scale decomposed application components – and can do so in a code-centric manner that promotes repeatability, resilience, and minimization of vendor lock-in.

2

Data is your biggest asset and needs to be prioritized. Ensure your digital development architecture, approach and resources are aligned to a comprehensive and adaptive data strategy.

3

Build multi-disciplined development team structures, embedded in the business around immediate requirements. Adopt key principles including ideation and decisions to promote speed - and technology guardrails to ensure the right level of control.

4

Ensure your teams have the correct ecosystem to support autonomy and easy collaboration for speed – including streamlined processes, automated governance, and a safe and accessible environment to build and iterate in development.

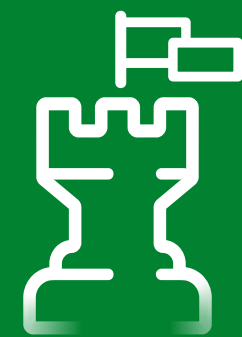
5

Make your access to innovation expertise highly available and adaptable – whether this is through technologies that remove barriers for the less digitally-skilled, building proficiency and specialism by up-skilling internally, or working with partners to ensure 'skills on demand'.

The bigger picture:

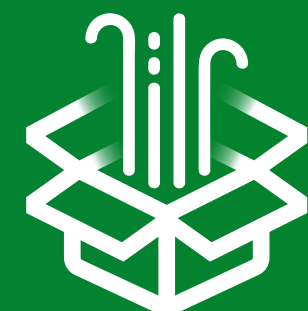
Creating your adaptive organization

Building services faster is just one crucial component of creating your adaptive organization of tomorrow. There are four other important areas for continuous transformation which allow you to build resilience, responsiveness, and relevance for the future:



Protected foundations

For a safe and secure digitally-enabled business



Optimize cost and agility

For efficiency today and flexibility for the future



Enhance effectiveness

For intelligent decisions and rapid action



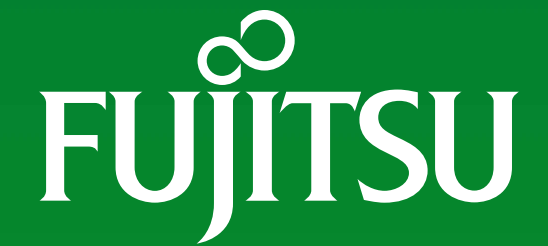
Build services faster

For delighting customers and disrupting competitors



Drive insight and new value

For business, consumers, and societal good



Accelerate and evolve holistically with Fujitsu

Every business has the potential to become an agile, adaptable, and thriving entity. With the support, expertise, and experience of Fujitsu, you can turn that potential into reality.

Get in touch and get started at
www.fujitsu.com/global/services/ao

FUJITSU

Tel: +44 (0) 123 579 771
Email: ask.fujitsu@uk.fujitsu.com
fujitsu.com/global

Reference: 4081

© Fujitsu 2021, the Fujitsu logo, are trademarks or registered trademarks of Fujitsu Limited in Japan and other countries. Other company, product and service names may be trademarks or registered trademarks of their respective owners. Technical data subject to modification and delivery subject to availability. Any liability that the data and illustrations are complete, actual or correct is excluded. Designations may be trademarks and/or copyrights of the respective manufacturer, the use of which by third parties for their own purposes may infringe the rights of such owner. Ask Fujitsu ID: **4081** All rights reserved.