



6 facteurs clés pour réussir sa démarche DevOps

DÉCOUVREZ LE RETOUR D'EXPÉRIENCE
DE NOS **CONSULTANTS DEVOPS MERITIS**

— Nicolas Marot — Cédric Leblond — Romain Dartigues



SOM- MAIRE



————— **PRÉSENTATION DES
CONTRIBUTEURS DU LIVRE BLANC**

————— **INTRODUCTION**

FACTEUR #1 : MODÉLISER LES PROCESSUS

**FACTEUR #2 : CONSTRUIRE UNE CULTURE
COLLABORATIVE**

**FACTEUR #3 : CONCEVOIR UN RÉFÉRENTIEL
COMMUN**

FACTEUR #4 : MESURER LA PERFORMANCE

**FACTEUR #5 : SÉLECTIONNER LES OUTILS
ET LES MÉTHODOLOGIES**

**FACTEUR #6 : INTÉGRER LE FEEDBACK
DES UTILISATEURS MÉTIERS**

————— **CONCLUSION**

————— **RESSOURCES**

————— **QUI SOMMES-NOUS**

————— **CONTACT**

3

4

5

6

7

8

9

10

11

12

13

14

PRÉSENTATION DES CONTRIBUTEURS DU LIVRE BLANC

— Nicolas Marot, Consultant DevOps et Agile chez Meritis

Nicolas Marot, consultant DevOps et Agile chez Meritis depuis 2 ans, est spécialisé en Infrastructure : gestion de configuration, déploiement continu, Cloud hybride, systèmes distribués, orchestration de conteneurs, sécurité.

Engagé depuis 2005 auprès de ses clients dans la réorganisation et l'optimisation d'équipes d'exploitation et d'hébergement, il contribue également activement à l'excellence opérationnelle des plateformes et services par son expertise technique, l'adoption des bonnes pratiques et le partage de ses connaissances.

Il occupe indifféremment les rôles d'Ingénieur Cloud ou de Product Owner Scrum au service de grands comptes ou de start-ups.

Passionné de nouvelles technologies, il consacre une partie de son temps libre à la veille technologique et à l'expérimentation de projets open source.

— Cédric Leblond Consultant DevOps et Azure chez Meritis

Cedric Leblond, consultant DevOps et Azure chez Meritis depuis 6 ans, est spécialisé en Software engineering (qualité logicielle, production et livraison).

Il accompagne les entreprises dans la modernisation de leurs pratiques de développement, et se concentre depuis 2008 sur les sujets de « Continuous Delivery » ainsi que la mise en place de plateformes.

Il est également un contributeur très actif au sein de la communauté DevOps de Meritis, et assure depuis de nombreuses années une veille technologique et un partage de ses connaissances. (.Net Ranger, co-organisateur FlowCon, animateur AZUG FR, MVP Azure).

— Romain Dartigues Consultant DevOps et administrateur systèmes Linux chez Meritis

Consultant Meritis et administrateur systèmes Linux DevOps chez ANTYAS depuis 2016.

En plus de 20 ans, il a pratiqué divers métiers des technologies de l'information, ce qui lui permet aujourd'hui **d'appréhender les difficultés et vocabulaires des équipes et projets. Il promeut également les technologies Open Source.**

INTRODUCTION

Le DevOps semble avoir trouvé sa place en entreprise. En effet, 57 % du portefeuille applicatif sera géré en DevOps en 2023*. Mais concrètement, de quoi s'agit-il ? C'est en octobre 2009 à l'occasion de la première conférence « DevOpsDays » organisée à Gand, en Belgique, que Patrick Debois, membre de la mouvance agile, emploie pour la première fois le terme DevOps. Défini de façon moins formelle que la méthode agile, la démarche DevOps a été « conçue » en vue de favoriser l'adoption des pratiques agiles par les métiers. Pour y parvenir, l'idée est de fluidifier les interactions entre développement et exploitation dans un but commun : délivrer de la valeur plus rapidement.

Ainsi, le DevOps n'est pas une norme ni un recueil de bonnes pratiques. Le DevOps est avant tout une philosophie, une méthodologie, une organisation, un modèle centré sur la capacité à faire travailler ensemble des profils totalement opposés. S'il n'existe pas une voie unique pour le mettre en œuvre, le concept n'en repose pas moins sur un certain nombre de principes fondateurs applicables à tous les secteurs : automatisation, responsabilisation, amélioration continue et centralisation des ressources dans un référentiel commun.

Si l'automatisation est une manière parmi d'autres d'aborder le DevOps, c'est bien le service qui est le premier vecteur de valeur, celui qui va permettre de guider les initiatives en interne comme le souligne Nicolas Marot, Senior DevOps Consultant chez Meritis PACA :

« Le point essentiel de la démarche DevOps, c'est le service. Chaque périmètre du projet doit fonctionner en mode service, c'est-à-dire proposer la valeur délivrée sous forme de services comme des machines virtuelles à la demande, l'intégration continue as a service, le DNS as a service... C'est le concept du "tout as a service". Cette approche service contribue pleinement au succès d'une démarche DevOps dans une organisation complexe. Mais elle prend du temps. »

En effet, la mise en œuvre du DevOps ne s'effectue pas du jour au lendemain. Selon la taille de l'entreprise et sa maturité sur le sujet, elle peut prendre en moyenne entre 2 et 5 ans et implique une remise en question permanente. Pour vous accompagner, nos consultants vous présentent quelques facteurs clés issus de leur expérience terrain.

*Source : DevOps : **Une transformation organisationnelle, humaine et technologique**, IDC, 2019



FACTEUR #1

MODÉLISER LES PROCESSUS

Les multiples itérations et livraisons inhérentes au principe de DevOps ne sont pas sans générer certains freins et points de frictions potentiels. En ce sens, la modélisation de l'ensemble des processus et l'identification des optimisations favorisent une certaine fluidité tout au long de la chaîne de valeur.

Pourquoi modéliser ?

Interactions avec les équipes internes non-DevOps, organisation en silos, résistance au changement, manque de clarté de certains objectifs, flou autour du concept DevOps, etc. Autant de freins qui peuvent limiter l'intérêt et la vélocité d'une démarche DevOps. En ce sens, plus les processus et les responsabilités seront clairement définis en amont, plus les projets pourront être menés efficacement de bout en bout.

Voilà pourquoi la modélisation est une clé de succès. Elle permet de fournir à l'ensemble des parties prenantes une vue claire sur les rôles de chacun, les livrables attendus, les points d'optimisation et les processus de livraison. Cette compréhension globale du chemin à emprunter et du but à atteindre réduit également le risque de résistance au changement au sein des équipes.

Comment modéliser ?

La modélisation repose sur plusieurs facteurs, selon la taille de l'entreprise et sa maturité. Pour Nicolas Marot, il existe deux façons de procéder à la transformation DevOps :

- Par une approche top down qui consiste à réorganiser les équipes, renommer les rôles au sein de l'organigramme. Les métiers quant à eux ne changent pas en profondeur.
- Par une approche Bottom up qui favorise l'expérimentation sur un périmètre bien défini. Dans ce cas-ci, il est essentiel de préserver au maximum le projet des dépendances techniques externes.

« Une fois la méthode établie, vous pouvez choisir les meilleurs candidats au sein de votre organisation pour commencer la transformation. Mais pour réussir votre projet DevOps, vous devez procéder étape par étape, conseille Nicolas Marot. Il est essentiel de ne pas brûler les étapes et que votre équipe ait déjà une bonne maturité Agile et une sensibilité DevOps. »

★ L'avis de l'expert

Romain Dartigues, consultant DevOps et administrateur systèmes Linux chez Meritis
« Un projet DevOps peut en effet intégrer plusieurs points de friction humains. Il est donc nécessaire d'identifier en amont tout ce qui peut être sujet à tension pour que chacun puisse en prendre connaissance et faciliter la communication tout au long du projet. Ce point est le premier facteur de succès d'une démarche DevOps. Donc plus les processus seront modélisés, plus les points d'étape pourront être optimisés. »

Entre 2014 et 2019, le nombre de collaborateurs travaillant au sein d'équipes DevOps est passé de

16 À 27 %

Source : *Enquête Dora – Google Cloud, Accelerate State of DevOps, 2019*

FACTEUR #2

CONSTRUIRE UNE NOUVELLE CULTURE COLLABORATIVE

La collaboration est intrinsèque au DevOps dont le principe consiste à associer au sein d'une même équipe deux métiers antagonistes : les développeurs d'un côté et les opérationnels de l'autre. Cette culture du partage est un facteur majeur dans la réussite du projet. La communication et la collaboration sont essentiels en vue d'assurer le bon déroulé du projet de A à Z.

Pourquoi déployer une culture collaborative ?

Le DevOps provenant directement de l'agile, il peut être sujet à une certaine forme de standardisation et de monétisation. En effet, de nombreux ouvrages de méthodologie tendent à faire croire qu'il existe une recette miracle pour réussir sa démarche DevOps et qu'il est possible de « certifier » une méthodologie particulière.

Or il n'en est rien. Le DevOps n'a pas vocation directement à garantir des bénéfices financiers ou un ROI. Le DevOps n'est pas une norme. Les outils, s'ils sont essentiels, n'en restent pas moins secondaires. Le DevOps repose sur la faculté à pouvoir travailler avec des méthodologies opposées mais pour un objectif commun : livrer de la valeur. D'où la nécessité de mettre en œuvre une culture collaborative.

Comment construire cette culture ?

Comme pour tout projet centré sur l'humain (Human Centric), le déploiement d'une nouvelle culture ne peut se faire du jour au lendemain. La clé de la collaboration repose avant tout sur la responsabilisation ! Nicolas Marot, a identifié 3 facteurs essentiels au déploiement d'une culture de la collaboration :

- La responsabilisation : « C'est la principale différence entre une méthodologie classique et le DevOps établi sur la base du principe *You build it, you run it*. Cette responsabilisation est une véritable clé de collaboration. » Autre atout : elle favorise également la transparence et la confiance !
- Le team building.
- La capacité à mettre en œuvre des flux de travail qui font intervenir plusieurs collaborateurs les incitant à travailler ensemble : revue de code, ateliers de conception, etc.

★ L'avis de l'expert

Romain Dartigues, consultant DevOps et administrateur systèmes Linux chez Meritis

« Le DevOps est avant tout une culture établie sur l'échange et non sur des outils. Elle implique notamment de disposer en interne d'une définition et d'une vision claires de la notion d'agilité. Lors d'une mission en entreprise, j'ai ainsi été confronté à deux équipes : l'une était composée de jeunes développeurs qui pratiquaient le Scrum et l'autre, de développeurs AS / 400 seniors qui fonctionnaient en Kanban. Alors qu'ils étaient censés être moins agiles que la 1re équipe, ils effectuaient 1 800 déploiements / an. Voilà pourquoi l'humain et la collaboration prévalent sur les outils et autres méthodologies que vous pourrez mettre en place ! »

La collaboration représente près d'un tiers

32 %

des actions prioritaires mises en œuvre pour soutenir les initiatives DevOps.

Source : *DevOps : une transformation organisationnelle, humaine et technologique*, IDC, 2019

FACTEUR #3

CONCEVOIR UN RÉFÉRENTIEL COMMUN

Comment des profils antagonistes peuvent-ils mener avec succès un projet s'ils sont incapables de comprendre de quoi on parle ? En adoptant un langage et un référentiel communs ! Un document de référence que l'ensemble des acteurs, développeurs ou métiers, pourraient consulter tout au long du projet pour s'assurer de bien comprendre ce dont il s'agit et ce qu'on attend d'eux. Une transparence indispensable pour réduire considérablement les risques d'erreur et d'incompréhension.

Pourquoi un référentiel commun ?

Nombre d'organisations fonctionnent aujourd'hui encore en silos. Afin de faciliter leur transformation, elles utilisent souvent une organisation matricielle. Toutefois cette matrice peut vite devenir complexe dès lors qu'elle intègre un trop grand nombre de dimensions. Idéalement elle doit en contenir 2 (par exemple, les dimensions hiérarchiques et projets, mais tout dépend du contexte), voire 3. Mais au-delà, elle devient trop compliquée à gérer. Pour éviter ce niveau de complexité, il est donc essentiel de définir précisément les termes utilisés et les processus au sein d'un référentiel commun.

★ L'avis de l'expert

Cédric Leblond, Consultant ALM, DevOps, Agile chez Meritis

« La mise en place d'un référentiel commun fait partie de tout projet DevOps. Il est important que chacun connaisse son rôle, son périmètre de responsabilités, et de comprendre comment les choses vont être orchestrées. Il faut donc un cadre clair pour que chacun puisse être autonome et prendre des initiatives. Ce référentiel commun peut intégrer des matrices de responsabilité, et tout élément concret qui permette de documenter le plus précisément possible les processus : qui intervient à quelle étape, quand mettre en œuvre l'automatisation... Tous les termes doivent être explicités ! Mais à chaque organisation, son référentiel. L'important est de savoir comment procéder. »



Une mauvaise communication est impliquée dans

19 %

des échecs des projets IT. Soit un projet sur trois dont l'échec est directement lié au facteur humain.

Source : 9^e étude Pulse of the profession 2017, réalisée par PMI (Project Management Institute)

Comment définir un référentiel commun ?

Le principe est extrêmement procéduré et codifié pour pouvoir automatiser au maximum les processus. Le facteur de succès repose alors sur la pertinence du choix des processus automatisés pour fluidifier les flux de travail.

Toutefois, comme le rappelle Romain Dartigues, mettre en œuvre un langage commun reste difficile. Il est alors possible de se reporter au principe ITIL et à ses ouvrages de référence. Il importe néanmoins de vérifier en amont que l'ITIL est bien compatible avec l'agilité.

FACTEUR #4

MESURER LA PERFORMANCE

Revue de sprint, bugs, déploiements, remontée des incidents... Les opportunités ne manquent pas de mesurer la performance tout au long d'un projet DevOps. Objectif : ancrer les processus dans un cycle d'amélioration continue en partageant les résultats d'analyse au sein de l'équipe. Il importe alors de définir et de mettre en place les indicateurs adaptés. Reste à savoir lesquels choisir.

Pourquoi mesurer la performance ?

Au sein d'une démarche DevOps, deux facteurs de performance entrent en jeu : l'un purement technique concerne la qualité des services délivrés et l'autre résolument humain. Lequel mesurer ? Les deux, tant la transition DevOps peut être longue. C'est pourquoi il est indispensable de pouvoir surveiller toute la chaîne de valeur et d'ajuster les processus le cas échéant. En revanche, dans une démarche DevOps, il n'existe pas de KPI ni de ROI à proprement parler.

La mesure de la performance intervient davantage comme « *une supervision de l'état d'esprit et de l'adoption des équipes que comme une mesure concrète* », précise Romain Dartigues. D'un point de vue technique, cette supervision des performances est importante pour permettre d'automatiser à tout moment. À la clé : la possibilité de savoir rapidement si le projet a réussi ou échoué.

Comment mesurer la performance ?

Pour suivre concrètement les performances et les améliorations apportées, les indicateurs à mettre en place varient selon le service concerné.

Cédric Leblond, partage quelques indicateurs types :

- ➔ Pour un service web, il peut s'agir du temps de réponse, du nombre d'erreurs, mais également de la disponibilité ou la cohérence des données.
- ➔ Vous pouvez aussi utiliser les mesures déjà en place au sein de l'ITIL comme la qualité de service, le temps de réparation d'un bug, le délai de remise en service...
- ➔ Certains indicateurs permettent de mesurer où vous en êtes par rapport aux objectifs définis : livraisons effectuées vs attendues, nombre de bugs détectés / incidents en production...
- ➔ Enfin, les indicateurs peuvent également être plus pragmatiques et liés à la durée des déploiements, au nombre d'incidents, au taux de disponibilité de l'infrastructure, etc.

Définir des indicateurs de pilotage et d'évaluation communs aux équipes DevOps est identifié comme le

2^{ème}

facteur de déploiement d'une culture agile dans l'entreprise.

Source : *Les initiatives DevOps en France, tendances et chiffres clés*, IDC, 2018



L'avis de l'expert

Nicolas Marot, Senior DevOps Consultant chez Meritis PACA

« Il est extrêmement important que le service soit bien supervisé par le biais des bons indicateurs. Il est donc essentiel de consacrer du temps à créer les indicateurs et à les faire évoluer dans le temps. C'est de l'amélioration continue. Voilà aussi comment vous pouvez vérifier les engagements pris quant au niveau de service attendu. C'est très structurant en termes de coûts et d'effort. »



FACTEUR #5

SÉLECTIONNER LES OUTILS ET LES MÉTHODOLOGIES

Monitoring, supervision de la performance, automatisation des processus, sécurité, communication, partage d'informations, etc. Les outils et méthodes ne manquent pas pour accompagner la démarche DevOps. Toutefois, difficile d'imposer des outils particuliers aux équipes. En effet, le choix des outils dépend à la fois des besoins, des compétences et même de la culture de l'entreprise.

Pourquoi mettre en place des outils ?

Dans une démarche DevOps, les outils sont au service de l'humain et non l'inverse. Donc attention de ne pas être trop dépendant des solutions déployées. Une des clés de réussite repose donc sur l'autonomie de chaque équipe à s'outiller comme elle le souhaite. En effet, l'uniformité ne favorise pas nécessairement l'agilité nécessaire au DevOps. Gestion de code, intégration continue, monitoring, déploiement continu... Si l'équipe est contrainte dans son choix, elle risque fort de perdre en efficacité et en productivité.

C'est pourquoi, comme l'explique, Romain Dartigues, la majorité des membres de l'équipe doit adhérer à l'outil et il est indispensable de s'en assurer au préalable ! Pour ce faire, il peut être intéressant de le faire tester par les développeurs, mais aussi par les métiers comme le marketing moins habitués à manipuler certains outils.

Comment choisir les bons outils et les bonnes méthodes ?

Nicolas Marot présente certains des outils et méthodes parmi les plus utiles et utilisés en DevOps :

- La méthode agile est celle qui convient le mieux au contexte DevOps. En fonction du livrable, le Scrum sera plus adapté à une démarche de développement de bout en bout alors que le Kanban répondra davantage à des processus moins réfléchis en amont.
- Côté outil, le Value Stream Mapping (VSM) va aider à définir toute la chaîne de valeur intégrant les différentes étapes depuis l'émergence d'une idée jusqu'à la mise en production d'une nouvelle fonctionnalité. Le VSM va décrire l'ensemble des processus, des intervenants et des actions mises en œuvre. Cette cartographie va alors permettre de définir les pistes d'optimisation les plus pertinentes et d'élaborer les processus à venir.
- Enfin, côté communication, les tableaux de bord ou dashboards favorisent un partage efficace et transparent de l'avancée du projet. Pour compléter, la messagerie instantanée et l'automatisation de type chatops ont l'avantage de pouvoir facilement s'intégrer à l'écosystème en place.

★ L'avis de l'expert

Cédric Leblond, Consultant
ALM, DevOps, Agile chez
Meritis

« Les outils Cloud Native émergent et poussent le modèle DevOps. Kubernetes par exemple, né du DevOps, facilite les processus. La plateforme est nativement ancrée dans cette idée d'automatisation. Les outils de surveillance Cloud Native sont eux aussi très adaptés à la démarche. Automatiquement remontée dans l'outil de monitoring, la mesure de la performance permet notamment d'orienter les efforts sur les améliorations à apporter aux processus. En ce sens, l'outil se met véritablement au service de l'humain et s'impose comme une clé de succès de la démarche DevOps. »

Tester de nouvelles
méthodes et outils basés
sur l'IA est une action
prioritaire pour

28 %

des équipes DevOps.

Source : IDC, Western Europe
DevOps Survey 2019

FACTEUR #6

INTÉGRER LE FEEDBACK DES UTILISATEURS MÉTIERS

Comme toute démarche agile, le DevOps doit placer l'utilisateur au centre des processus pour fonctionner. En intégrant le feedback tout au long du développement, il est ainsi possible d'ajuster en permanence (ou presque) les objectifs au regard des besoins utilisateurs et même d'anticiper les possibles écueils à venir. En ce sens, le feedback utilisateur est une clé d'agilité.

Pourquoi intégrer le feedback des utilisateurs métiers ?

Les feedbacks utilisateurs permettent d'accélérer les processus en permettant de vérifier si la fonctionnalité remplit sa fonction et de la mettre à disposition beaucoup plus rapidement que dans le cas de campagnes de tests purement techniques. Et ce d'autant plus lorsqu'il s'agit d'intégrer le retour de plusieurs milliers voire millions d'utilisateurs. Le but de toute démarche DevOps est de mettre à disposition des produits qui fonctionnent donc autant s'en assurer le plus tôt possible dans la chaîne de processus.

Autre facteur de succès : le feedback utilisateur permet également de prendre en compte les « ressentis » impossibles à mesurer par des indicateurs. Cette capacité s'avère essentielle pour améliorer le service, tant par sa qualité que par ses performances. Pour la marque, c'est aussi la possibilité de reprendre certains arguments de vente issus des utilisateurs eux-mêmes à des fins commerciales.

★ L'avis de l'expert

Nicolas Marot, Senior DevOps Consultant chez Meritis PACA

« L'agile nous donne la réponse. Le DevOps est conçu pour les utilisateurs finaux et non pour l'entreprise elle-même. Dans le cas d'une équipe DevOps très mature, on peut envisager la collecte automatique des parcours utilisateurs lors de la mise à jour d'une app mobile, identifier ainsi les fonctionnalités moins performantes ou défectueuses et, dans certains cas, annuler automatiquement la mise à jour. »



En 2021,

35 À 40 %

des projets de développement applicatif seront effectués en DevOps en France. D'où l'importance croissante du feedback utilisateur métier.

Source : Les initiatives DevOps en France, tendances et chiffres clés, IDC, 2018

Comment intégrer le feedback des utilisateurs métiers ?

Cette démarche de vérification est essentielle afin de vérifier régulièrement que le travail est correctement effectué. Toutefois, il n'est pas si simple à mettre en œuvre au sein des organisations et certains projets n'impliquent pas nécessairement de recueillir le feedback.

- Toutefois, dans le cas où cette collecte serait utile, deux options principales s'offrent aux équipes DevOps :
- ➔ Intégrer les utilisateurs métiers directement au sein même des équipes DevOps. Le feedback peut alors s'effectuer par des échanges directs ou via des outils de collaboration et de communication.
 - ➔ Ou à travers des rapports d'utilisation et des comptes rendus de tests utilisateurs. Dans ce cas, il est essentiel de partager les résultats à l'ensemble des membres de l'équipe DevOps pour une transparence totale et une optimisation à tous les niveaux.



CONCLUSION

Nous avons fait le choix de nous restreindre à six facteurs mais la liste est loin d'être exhaustive. Il en existe d'autres, tous valables. **Ce qui compte, c'est de trouver ceux qui conviennent à votre entreprise, à votre culture, à votre organisation et, le plus important, à vos équipes.** Et attention aux pièges ! Ce n'est pas parce que vous faites du DevOps, que vous allez fonctionner en mode agile du jour au lendemain. **Le DevOps implique du temps et des ressources.**

Si le recours au DevOps s'est considérablement accéléré ces dernières années, comme le rappelle notre consultant Romain Dartigues, il reste difficile de faire du DevOps dans un monde qui ne l'est pas et au sein d'entreprises qui, quels que soient leur taille et leur niveau de transformation, n'en demeurent pas moins encore silotées, notamment du côté des directions métiers ou générales.

Quel est alors le premier facteur de succès de la démarche ? L'humain ! C'est pourquoi la proximité et la communication au sein des équipes sont clés. La proximité pour favoriser l'empathie et les décisions rapides, et la communication pour faciliter les échanges entre les équipes DevOps et non-DevOps et s'assurer que le projet avance correctement. C'est là que les outils interviennent pour automatiser, fluidifier et accélérer les processus.

En ce sens, la taille des équipes est importante et s'impose, elle aussi, comme un facteur de succès. Il est souvent recommandé d'intégrer moins de vingt collaborateurs dans une équipe DevOps pour limiter le risque de ralentir la prise de décisions. Rappelez-vous alors que le DevOps n'est pas un cahier des charges ISO. En cas de doute, c'est toujours vers l'humain qu'il sera nécessaire de revenir.

RESSOURCES

— Le blog de Meritis

DevOps, vous avez dit DevOps ?

— Études

- **Les initiatives DevOps en France, tendances et chiffres clés**, IDC, 2018
- **Western Europe DevOps Survey** 2019, IDC, 2019
- **DevOps : une transformation organisationnelle, humaine et technologique**, IDC, 2019

— Ouvrages de référence

- *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment*, de Jez Humble et David Farley, Éditions Addison-Wesley, juillet 2010, 512 p.
- *Continuous integration: Improving Software Quality and Reducing Risk*, de Paul M. Duvall, Éditions Addison-Wesley, 2007, 283 p.



QUI SOMMES-NOUS ?

— Meritis, le talent d'aller plus loin.

Conseil, pilotage et développement IT.

Meritis est une société de conseil en transformation des Systèmes d'Information et Organisations.

- Notre approche ? Accompagner nos clients sur l'ensemble de la chaîne de valeur : cadrage personnalisé, pilotage & développement applicatif pour les projets IT.
- Notre mission ? Connecter les meilleurs talents au service de la transformation numérique pour donner un temps d'avance aux entreprises.

Nos 600 consultants vous accompagnent avec agilité dans tous vos projets de transformation digitale. Un seul objectif : vous emmener plus loin.

EN SAVOIR PLUS

— Notre expertise DevOps

Mettre en place une culture DevOps de manière pérenne

Vous souhaitez :

- Réduire les tensions entre développeurs et administrateurs systèmes ?
- Assurer qualité et continuité de votre service informatique ?
- Raccourcir les délais de commercialisation de vos produits logiciels ?
- Lancer de nouvelles fonctionnalités plus rapidement ?
- Infuser la culture DevOps de manière pérenne ?

Meritis vous accompagne dans la mise en place des pratiques DevOps au sein de vos équipes et s'engage à INFUSER cette culture à travers :



— Audit

Nous réalisons un audit de vos processus et méthodes de fabrication logiciels (plan d'amélioration qualité)



— Déploiement

Nous mettons en place l'automatisation de vos tests (stratégie, conception, mise en place, méthodes et outils)



— Organisation

Nous coordonnons vos équipes Software Factory et DevOps Infra à travers Kanban, roadmaps techniques, suivi KPIs, construction des budgets etc. (gestion de projet)



— Formation

Nous accompagnons les équipes dans leurs changements de pratiques, et travaillons sur un inventaire précis des nouvelles ressources utilisées et leur formation associée. (Accompagnement au changement)

— Nous contacter

Un projet, une question, vous souhaitez en savoir plus ?

Contactez-nous ! Nos équipes d'experts sont à votre disposition pour répondre à toutes vos questions.

NOUS CONTACTER

NOUS REJOINDRE

in

f



meritis.fr



meritis