

Sommaire

Sommaire 2

Introduction 3

Présentation de l’entreprise 3

problématique 3

Partie 1: compréhension du besoin et la traduction technique de la réponse apportée. 4

KPI, SLI et SLA dans la Méthodologie de Projet 4

la Modernisation du Workflow 5

Bibliographie 6

Introduction

Dans un paysage technologique et commercial en mutation constante, les organisations doivent revoir leur approche du développement et du déploiement de logiciels. L'adoption de l'approche DevOps offre une solution intégrée pour garantir l'efficacité, la qualité et la stabilité des systèmes logiciels, tout en favorisant une culture d'innovation et de collaboration.

Ce dossier examine l'application des principes DevOps dans une entreprise spécialisée dans l'IoT. Nous analyserons les défis rencontrés par cette entreprise et explorerons des solutions pour atteindre ses objectifs de modernisation.

Nous mettrons en lumière les besoins spécifiques de ce projet, soulignant l'importance de l'approche DevOps pour encourager la collaboration, optimiser les processus de développement et de déploiement, réduire les risques et favoriser l'agilité face au changement.

À travers cette étude, nous proposerons des recommandations et des stratégies pour une mise en œuvre efficace de l'approche DevOps, adaptée aux objectifs et aux besoins uniques de l'entreprise. Cela facilitera son développement continu et son succès dans un environnement concurrentiel en perpétuelle évolution.

Présentation de l’entreprise

L'entreprise se spécialise dans le développement d'instruments de diagnostic dédiés au contrôle de la qualité. Elle se distingue par la fourniture de produits de haute qualité et jouit d'une solide réputation dans les domaines de la mesure de l'air, de l'eau, du sol et de la recherche sur la biodiversité.

Elle a conçu un logiciel de bureau centralisé pour la gestion unifiée de ses produits, simplifiant ainsi la configuration, la mise à jour et l'étalonnage des instruments. Récemment, l'entreprise a initié le développement d'une application mobile permettant aux clients de configurer leurs appareils via leurs smartphones Android et iOS.

Le langage de développement privilégié par l'entreprise est principalement TypeScript et JavaScript, tandis que React Native et Electron assurent respectivement l'accès web et la création d'applications mobiles et de bureau.

En permanence à la recherche de nouveaux talents, elle recherche particulièrement des développeurs JavaScript full-stack seniors, dotés d'une expertise approfondie en React Native et Electron. Structurée en cinq équipes de développement, chacune dirigée par un product owner, un scrum master, un lead développeur et un QA.

L'équipe DX (Developer Experience) est chargée de mettre en place des outils de développement, de dispenser des formations et de promouvoir les meilleures pratiques.

problématique

De nombreux petits projets sont stockés dans des repository distincts. Certaines fonctionnalités requièrent des modifications dans plusieurs repository. L'objectif est de gérer ces modifications de manière centralisée et de les déployer de façon coordonnée.

Actuellement, lorsqu'un repository est modifié, une pull request est soumise, suivie de l'attente de la validation, puis d'une release, et encore d'une validation. Ensuite, le déploiement est effectué, suivi d'une nouvelle attente de validation. Ce processus doit ensuite être répété pour les autres repository impactés par la modification. Par exemple, si le firmware du capteur d'air est modifié, les ajustements doivent également être apportés à l'application mobile, à l'application de bureau, au back-end et au front-end web.

Des difficultés émergent également dans l'intégration des développeurs sur la partie front-end. Certains réclament des documentations ou des tests, mais le QA ne réalise que des tests physiques et n'a pas le temps d'effectuer des tests automatisés. Sur le back-end et l'électronique, des tests unitaires sont utilisés, mais rien n'est fait pour le front-end.

Enfin, l'analyse des erreurs rencontrées par les utilisateurs pose problème. Malgré l'efficacité de nos outils de monitoring, nous souhaitons obtenir des informations plus précises sur le code défectueux, les retours d'erreurs, etc. Des cas liés aux versions de navigateurs, systèmes d'exploitation et firmwares ainsi que des problèmes de connexions nécessitent des informations plus détaillées.

Partie 1: compréhension du besoin et la traduction technique de la réponse apportée.

Pour répondre à la problématique soulevée, voici notre proposition de solution.

KPI, SLI et SLA dans la Méthodologie de Projet

La mise en place d'une méthodologie efficace constitue un pilier fondamental pour le succès des projets, tant dans leur gestion que dans leur exécution. Dans ce contexte, la définition précise des indicateurs clés de performance (KPI[[1]](#footnote-1)), des objectifs de niveau de service (SLI[[2]](#footnote-2)) et des accords de niveau de service (SLA[[3]](#footnote-3)) revêt une importance cruciale. Ces éléments offrent une feuille de route permettant de mesurer, d'évaluer et de garantir la qualité, la fiabilité et la performance des produits ou services développés.

Dans le cadre du projet, axé sur la création d'outils de diagnostic dans le contrôle qualité, l'importance des KPI, SLI et SLA est particulièrement accentuée. Car la société, reconnue pour son expertise dans la mesure de la qualité de l'air et d'autres domaines connexes, s'engage à fournir des solutions de haute qualité à ses clients. Ainsi que, dans un contexte DevOps, où l'accent est mis sur l'automatisation, la collaboration et l'amélioration continue entre les équipes de développement et d'exploitation, la définition et la mesure des KPI, SLA et SLI sont cruciales pour garantir la qualité, la fiabilité et la performance des services fournis.

Pour le calcul des KPI pour le SLI et le SLA, il est important de choisir des mesures qui reflètent la performance et la disponibilité des services offerts par les applications développées. Ainsi dans le cadre du projet, nous proposant différentes mesures récupérés pour le calcul de KPI à partir de SLA et SLI:

SLI

1. Taux de Disponibilité :

Mesuré à partir du temps où les services sont disponibles pour les utilisateurs finaux.

1. Temps de Réponse Moyen :

Calculé à partir du temps écoulé entre la réception de la requête et l'émission de la réponse.

SLA

1. Objectif de Disponibilité :

Définit le niveau de disponibilité attendu pour les services, généralement exprimé en pourcentage.

1. Temps de Résolution des Incidents :

Spécifie le délai maximal acceptable pour résoudre les incidents ou les problèmes signalés.

En prenant en compte ces indicateur nous pourrons soit calculer un KPI (dans le cas ou on choisi une donnée pour SLI et une donnée pour SLA), soit deux. La pertinence de choix doit être discuter en équipe, afin de instaurer une vision général des attentes au sein de l’équipe.

La formule basique pour calculer le KPI pourrait être :

Cette formule exprime le rapport entre le niveau de service réel (SLI) et le niveau de service attendu (SLA), généralement exprimé en pourcentage pour obtenir une mesure comparative plus intuitive.

la Modernisation du Workflow

Pour commencer, nous pensons que pour atteindre le premier objectif qui est

L'objectif est de gérer ces modifications de manière centralisée et de les déployer de façon coordonnée.

Bibliographie

A successful Git branching model. (s. d.). nvie.com. https://nvie.com/posts/a-successful-git-branching-model/

Atlassian. (s. d.). Principes des microservices | Atlassian. https://www.atlassian.com/fr/microservices

Benefits of a DevOps environment | MuleSoft. (s. d.). MuleSoft. https://www.mulesoft.com/resources/api/devops-environment-benefits

Bogdan Stashchuk. (2021a, novembre 4). Back-end development and APIs - FreeCodeCamp tutorial [Vidéo]. YouTube. https://www.youtube.com/watch?v=hHLmb3OD7Mo

Bogdan Stashchuk. (2021b, novembre 4). Back-end development and APIs - FreeCodeCamp tutorial [Vidéo]. YouTube. https://www.youtube.com/watch?v=hHLmb3OD7Mo

Catoire, L. (2023, 16 mai). Quelle architecture de projet choisir entre micro-services et monolithe modulaire ? Efficience IT. https://www.itefficience.com/article/quelle-architecture-de-projet-choisir-entre-micro-service-ou-monolithe-modulaire

DevOps Journey. (2023, 17 octobre). How to design a modern CI/CD pipeline [Vidéo]. YouTube. https://www.youtube.com/watch?v=KnSBNd3b0qI

Losoviz, L. (2023, 12 septembre). Mono-repo vs multi-repo : avantages et inconvénients des stratégies de dépôt de code. Kinsta®. https://kinsta.com/fr/blog/mono-repo-vs-multi-repo/

McKendrick, J. (2017, 20 avril). Pourquoi les microservices ne sont peut-être pas faits pour tout le monde. ZDNetFR. https://www.zdnet.fr/amp/actualites/pourquoi-les-microservices-ne-sont-peut-etre-pas-faits-pour-tout-le-monde-39851458.htm

Monolithic vs. Microservices : Why decoupled and headless architectures are the future | Contentstack. (s. d.). https://www.contentstack.com/cms-guides/monolithic-vs-microservices-cms-architectures

Wallen, J. (2022, 7 novembre). How to build a docker image and upload it to Docker Hub. TechRepublic. https://www.techrepublic.com/article/how-to-build-a-docker-image-and-upload-it-to-docker-hub/

What Is an API Gateway ? (s. d.). Nginx. https://www.nginx.com/learn/api-gateway/

1. KPI en anglais ou indicateur de performance en français est une mesure quantitative qui vous permet de suivre la progression de votre équipe ou organisation au regard de vos objectifs commerciaux clés. Dans cette mise à jour, nous verrons comment définir et suivre des KPI adaptés à votre activité. [↑](#footnote-ref-1)
2. Les SLI sont les indicateurs factuels de la disponibilité de votre service. Ils représentent l’état de votre service à un moment donné. Ce sont ces indicateurs que l’on mesurera avec le système de monitoring pour prouver le taux de disponibilité d’un service donné. [↑](#footnote-ref-2)
3. SLA est un engagement entre deux parties. Cela peut être entre deux entreprises (un provider et un client) ou bien entre départements dans une grande entreprise. Le SLA va définir votre engagement sur la disponibilité du service. [↑](#footnote-ref-3)