

Intégration et travail en équipe

Durée : 1 jour

Objectifs

Ce cours a pour objectif :

- Une bonne intégration des participants au groupe en mettant en valeur l'importance de réussir cette phase dans toute prise de fonction
- Une prise de conscience de l'importance de la dimension relationnelle et comportementale dans l'atteinte des objectifs pour les amener à accorder de l'attention dans la relation à des aspects comme l'écoute, la disponibilité, la confiance, l'esprit de coopération, etc.
- Une valorisation de l'impact de la méthodologie choisie sur l'atteinte des objectifs pour amener les participants à développer une approche structurée de leur travail.
- Un accompagnement à l'ancrage d'une projection individuelle et collective positive à l'issue du parcours.

Contenu

Accueil et Présentation du Parcours

- Présentation du Groupe Global Knowledge
- Présentation du programme du Parcours et des moments clés
- Présentation des règles générales d'organisation

Jeux pédagogiques et ludiques

Dans un cadre décontracté et une ambiance de jeu, les participants sont accompagnés tout au long de la journée à travers des jeux pédagogiques et ludiques pour les amener à s'intégrer et à prendre conscience de l'importance des dimensions non techniques et de leur impact sur l'atteinte de leurs objectifs et sur leur réussite dans leur rôle.

- Tirage au sort et organisation en équipes
- Jeu 1 : Plus tu connais plus tu gagnes
 - ✚ Principe : présentation croisée des participants
 - ✚ Objectifs pédagogiques : Se connaître les uns les autres en mettant en relief la richesse du groupe
- Jeu 2 : Les 5 objets
 - ✚ Principe : Transcription d'un schéma d'objets par commande distante
 - ✚ Objectifs :
 - Mettre en évidence l'importance de la phase de stratégie et planification dans tout projet
 - Analyser les causes possibles des écarts entre le message transmis et le message reçu
 - Valorisation de sens du détail et de la gestion optimisée du temps
- Jeu 3 : Le cercle des objets
 - ✚ Principe : Transmission de multiples objets au sein du groupe avec diverses contraintes
 - ✚ Objectifs pédagogiques :
 - Développement de la capacité à gérer la pression et le stress lié à la multiplicité des objets et à leur spécificité
 - Respect du récepteur dans l'acte de transmission et intégration de ses contraintes
 - Lien avec l'équilibre vie professionnelle/vie professionnelle d'une part et avec le renvoi des balles dans l'entreprise
- Jeu 4 : Le mur de la confiance
 - ✚ Principe : Aller vers un « mur de collègues »
 - ✚ Objectifs pédagogiques :
 - Prise de conscience des participants de leur distance spontanée de la notion de confiance quand le vis-à-vis en est incontestablement digne
 - Valoriser la dimension stratégie de l'équipe et son impact sur la performance individuelle et collective
- Jeu 5 : La corde au carré
 - ✚ Principe : Construction d'un carré parfait
 - ✚ Objectifs pédagogiques :
 - Renforcement de l'importance de l'organisation, de la planification et de la stratégie de groupe dans l'atteinte des objectifs
 - Mise en évidence des aptitudes du groupe à l'application minutieuse d'un plan de déploiement préétabli
 - Valorisation de la capacité du groupe à rebondir en cas d'imprévu
- Jeu 6 : La tour de papier
 - ✚ Principe : Construction de la plus haute tour
 - ✚ Objectifs pédagogiques :
 - Développement de la stratégie et de la différenciation
 - Accompagnement des participants à la valorisation du sens du détail et de la rigueur ainsi que leur

- impact sur la performance
- Renforcement du sens d'équipe et de l'importance de la planification et de l'organisation et la répartition des rôles
- Jeu 7 : Hymne du Groupe
 - ✚ Principe : Construire une chorégraphie d'équipe avec une chanson autour du métier
 - ✚ Objectifs pédagogiques :
 - Construction d'équipe
 - Activation des compétences de créativité des participants
 - Accompagnement des participants à la mise en avant des dimensions de leur futur métier
 - Valorisation du métier et projection positive des participants
 - Ancrage de l'émotion positive à travers la musique et la chorégraphie

Visualisation et débriefing

- La visualisation des différents exercices permet aux participants de prendre du recul face à leurs attitudes et comportement.
- Accompagnement vers une synthèse des enseignements de la journée autant à l'échelle individuelle que collective.
- Mise en avant de l'importance des notions suivantes :
 - ✚ Intégration au sein d'un nouvel environnement
 - ✚ Cohésion d'équipe
 - ✚ Travail en équipe
 - ✚ Stratégie et organisation du travail
 - ✚ Sens du détail et rigueur dans la mise en œuvre d'un plan d'action
 - ✚ Le récepteur est central dans l'acte de communication
 - ✚ Plaisir et travail

Le plus de cette formation

Dans un cadre convivial, les participants sont accompagnés à passer d'un ensemble d'individus à une équipe qui apprend dans le plaisir et qui se projette positivement.

Les bases du SQL

Durée: 3 Jours

Résumé:

SQL est le langage universel des bases de données relationnelles, et ce quel que soit le système de gestion de la base (Oracle, DB2 d'IBM, Access et SQL Server de Microsoft, MySQL, MariaDB, PostgreSQL de l'open source...). Langage incontournable dans les développements d'application, le SQL est aussi utilisé dans les outils d'analyse de données. Cette formation permet aux participants de comprendre la conception des bases de données relationnelles et d'acquérir les compétences et connaissances nécessaires pour manipuler le langage SQL.

Public visé:

Cette formation s'adresse aux développeurs, administrateurs, architectes de base de données et utilisateurs de la Business Intelligence.

Objectifs pédagogiques:

- Comprendre la structure des tables, les clés primaires et étrangères du modèle relationnel
 - Effectuer des mises à jour de données (création, modification et suppression de lignes), des créations de tables et de vues
 - Etre capable de faire des consultations, sur une ou plusieurs tables d'une base de données relationnelle, à l'aide du langage SQL
-

Pré-requis:

Pour suivre cette formation, il faut avoir des connaissances générales en informatique.

Contenu:

Principes et concepts du modèle relationnel

- Les domaines, les relations, les tables, les lignes et colonnes
- Les dépendances fonctionnelles, les clés (candidates, primaires)
- Les intégrités d'entité, les intégrités référentielles (clés étrangères), les valeurs NULL
- Normalisation des données : les formes normales
- Utilité de la normalisation pour la base et conséquences lors des accès
- La modélisation des données (MCD Modèle Conceptuel de Données, MLD (Modèle Logique de Données))

Le modèle relationnel dans la technologie Objet

- La modélisation avec les diagrammes de données d'UML (Unified Modeling Language)
- La correspondance entre le relationnel et l'objet (Mapping O/R)
- La persistance des données
- Le Design Pattern DAO (Data Access Object) et ses implémentations

Les requêtes avec le langage SQL

- Structure générale de l'ordre SELECT
- Sélectionner la table, les colonnes
- Sélectionner le résultat : restrictions avec la clause WHERE. Les opérateurs de restrictions
- Trier les résultats avec la clause ORDER BY

Accéder à plusieurs tables, les jointures internes : INNER JOIN. Les sous-requêtes

- Effectuer des statistiques avec les fonctions d'agrégation : COUNT, MAX, MIN, AVG, SUM
- Partitionner les résultats statistiques : GROUP BY et HAVING
- Les autres opérateurs ensemblistes (UNION, EXCEPT et INTERSECT)
- Accéder à plusieurs tables : les jointures externes
- Les fonctions SQL (manipulations de chaînes de caractères, manipulations de dates...)

La mise à jour de données avec le langage SQL

- Valider et invalider des mises à jour (Commit/Rollback)
- Créer des lignes dans une table : INSERT
- Modifier des lignes : UPDATE
- Supprimer des lignes : DELETE
- Conséquences des intégrités référentielles

Travailler avec des tables et avec des vues

- Présentation et intérêt des vues
- Différences entre table et vue
- Création et manipulation d'une vue avec SQL

Evolutions des données d'entreprises

- Du relationnel au décisionnel, l'intelligence décisionnelle ou la Business Intelligence
- Du décisionnel au Master Data Management (MDM)
- Le Big data

Travaux pratiques

- Schéma de la base
- Les requêtes simples – accès une table
- Les jointures
- Outer Join
- Sous-requêtes
- Fonctions d'agrégation - partitionnement
- Operations ensemblistes

Méthodes pédagogiques :

Un support de cours en français sera remis aux stagiaires. Plus de 50% de pratique. Lors des travaux pratiques, les participants analyseront une base de données relationnelle simple, afin de comprendre les concepts du modèle relationnel et la modélisation. Ils manipuleront le langage SQL, s'appliquant à toute base de données (Oracle, DB2, Access, SQL Server, MySQL, MariaDB, PostgreSQL...). Ils écriront des requêtes SQL permettant la consultation de données normalisées, manipuleront et analyseront les données de plusieurs tables. Ils écriront des requêtes de statistiques, des requêtes de mises à jour (création, modification et suppression de lignes). Et termineront leur apprentissage par la création de tables et de vues, afin d'être opérationnels sur l'ensemble des concepts du langage SQL et d'être capable d'aborder sereinement le développement d'application et/ou l'analyse de données.

Autres moyens pédagogiques et de suivi:

- Compétence du formateur : Les experts qui animent la formation sont des spécialistes des matières abordées et ont au minimum cinq ans d'expérience d'animation. Nos équipes ont validé à la fois leurs connaissances techniques (certifications le cas échéant) ainsi que leur compétence pédagogique.
- Evaluation : Chaque participant, à l'issue de la formation, répond à un questionnaire d'évaluation qui est ensuite étudié par nos équipes pédagogiques en vue de maintenir et d'améliorer la qualité de nos prestations.
- Suivi d'exécution : Une feuille d'émargement par demi-journée de présence est signée par tous les participants et le formateur.

La conception orientée objet

Durée : 2 jours

Résumé

Cette formation permet d'acquérir les connaissances de base solides sur la conception objet et celle basée sur les composants, en manipulant des outils de conception et de développement. Elle présente et démystifie le vocabulaire de cette technologie, souvent considéré, par les développeurs de l'approche procédurale, comme complexe.

Les participants seront capables de comprendre et de raisonner en classes et objets, d'appréhender la notation UML et la nécessité de la modélisation. Il s'agit d'assimiler les architectures et technologies objets avant d'entreprendre l'étude d'un langage objet ou des développements "orientés objets".

Objectifs

A l'issue de la formation, les stagiaires seront en mesure de :

- Connaître les concepts essentiels de l'Objet pour faciliter l'approche de tout langage de programmation Orienté Objet.
- Connaître les manipulations de base des outils de conception et développement (Eclipse, Visual Studio...).
- Appréhender la modélisation UML pour l'analyse fonctionnelle et technique.

Public visé

Développeurs, Concepteurs, Chefs de projets, Architectes techniques, Testeurs

Prérequis

Les participants doivent avoir des connaissances générales en informatique, notamment en matière de conception d'applications.

Pédagogie

Cette formation permet aux participants une mise en pratique progressive de tous les concepts objets, via un langage objet dans un outil de développement. Une application simple sera mise en œuvre afin d'aborder la conception et la modélisation objet et d'en comprendre les tenants et les aboutissants. Le participant sera ainsi opérationnel pour débiter dans le développement d'application et ce quel que soit le langage objet choisi, l'outil de développement et le type d'application à mettre en œuvre (Standalone, Web, mobile,...).

40% Théorie 60% de pratique

Contenu :

Objectifs de la conception Objet

- Historique de l'objet.
- L'objet par rapport à l'approche procédurale.
- Les principes fondamentaux de la conception orientée "objet".
- La nécessité de modéliser, la notation UML, les outils de modélisation.

Conception d'applications objets, l'industrie de l'objet

- Le vocabulaire : Objet – Composant – Framework
- Les Framework Java EE et .Net.
- Les langages Objet (Java, C++, C#, VB net, PHP...).
- L'objet et les bases de données relationnelles.
- Le modèle en couches dans les architectures n-tiers. Le design pattern MVC (Modèle – Vue – Contrôleur), les frameworks implémentant le MVC.
- Les objets métier, les composants.

- Le principe de la réutilisation des objets dans différentes applications (client lourd, client léger, client riche).
- Les outils de développement, les IDE (Integrated Development Environment) : Eclipse, NetBeans, Visual studio...

Les tests dans la conception d'applications objets

- Les tests avec les cas d'utilisation d'UML.
- Les outils de tests, l'intégration avec les outils de conception et de développement.
- Les outils de debug.
- Intérêt dans la conception de séparer le conceptuel (le modèle) du visuel (la présentation).
- **Travaux Pratiques** : Démonstration des outils, via la mise en œuvre dans l'application de gestion commerciale de la classe Client.

Classes – Objets - Instanciation

- Les packages et les espaces de noms, les bibliothèques.
- Les classes, les attributs, les méthodes (constructeurs, destructeurs et accesseurs).
- L'instanciation, les objets.
- **Travaux Pratiques** : Manipuler un outil de développement, écrire du code, mettant en œuvre la classe *Produit* de la gestion commerciale en respectant le pattern MVC.

Encapsulation

- Définition et intérêt.
- Visibilité des variables
- Mise en œuvre de l'encapsulation.
- **Travaux Pratiques** : Modifier la classe *Produit* en respectant l'encapsulation. Mettre en œuvre plusieurs constructeurs. Effectuer des tests. Faire évoluer le modèle de la gestion commerciale sans impact sur l'application cliente.

Liens entre les classes : Composition, agrégation

- Définition et utilité des liens de composition et d'agrégation.
- La représentation UML des liens dans le diagramme de classe.
- **Travaux Pratiques** : Conception de classes de composition et d'agrégation pour la gestion commerciale. Intégration des liens dans la classe *Produit*. Impacts sur la maintenance de l'application cliente.

Liens entre les classes : Héritage simple et multiple

- Définition et utilité des liens de généralisation/spécialisation.
- Le polymorphisme, la surcharge.
- La représentation UML des liens d'héritage.
- Les classes abstraites.
- Les références à l'objet.
- L'héritage multiple et les limites dans les langages Objet.
- **Travaux Pratiques** : Evolution de notre modèle objet, conception de deux classes filles, de la classe *Produit*. Manipulation des méthodes héritées, redéfinition d'une méthode de la classe mère dans une classe fille, utilité de méthode abstraite, maîtriser l'intérêt de l'héritage.

Les interfaces

- Définition et intérêts des interfaces.
- L'héritage multiple avec les interfaces.
- Les interfaces dans les architectures distribuées.
- **Travaux Pratiques** : Mettre en œuvre un héritage multiple sur les classes filles de notre *Produit*, en concevant et manipulant une interface, afin de comprendre le polymorphisme.

Travaux pratiques

Définition du produit

- Faire la représentation uml
- Définir le code java pour la classe produit
- Définir le code java pour tester la classe produit

Agrégation ou composition

- Faire la représentation uml
- Définir la classe dénomination
- Modifier la classe produit
- Modifier la classe testproduit

Héritage simple

- Faire le diagramme de classes
- Définir les classes *Produitperissable* et

Produitsaisonnier

- Créer la classe testproduits

Polymorphisme

- Modifier la classe produitsaisonnier
- Classe testproduits

Héritage multiple

- Création de l'interface saison
- Création de la classe produitfugace
- Modification de la classe produitsaisonnier
- Modification de la classe testproduits
- Création de la classe testsaisons

Autre(s) Information(s):

Un support de cours électronique en français sera remis aux stagiaires.

Introduction à Java

Durée : 2 jours

Résumé

Cette formation permet de prendre en main l'environnement JAVA.

Public visé

Administrateurs systèmes et réseaux.

Prérequis

Il est nécessaire d'avoir suivi la formation GKCOBJ « Les concepts : de la modélisation au développement objet » ou posséder les connaissances équivalentes.

Contenu :

Introduction Java, Java SE

- Les origines de Java, son historique
- Le JDK (Java Development Kit), ses versions
- L'outil de développement : Eclipse
- La plate-forme Java SE 8, Standard Edition
- Le fonctionnement de Java, la machine virtuelle (JVM), les exécutable, la gestion de la mémoire (garbage collector)
- La documentation, Javadoc

Le langage de programmation Objet de Java

- Les packages, les classes, les méthodes
- Les objets, les variables, les composants
- La création du projet Java sous Eclipse

Java EE : Les architectures logicielles et les techniques

- Architecture logicielle et technique en couches
- La plate-forme Java EE Enterprise Edition
- Les principaux composants Java EE (EJB, Servlet, JSP)
- Les containers et leurs utilités
- Java embarqué, Android

Les archives Java et le déploiement d'applications

- Les archives .jar, .war, .ear
- Les serveurs d'applications

L'architecture Web

- Mise en œuvre d'un projet Web
- Servlet – JSP – Java Bean
- Déploiement sous Apache - Tomcat

Travaux pratique

Manipulations de .jar
IHM – Client Lourd
gestion application Web Java EE (création – déploiement)

Autre(s) Information(s)

Un support de cours sera remis aux stagiaires

Les fondamentaux de la méthode Agile SCRUM

Durée: 2 Jours

Résumé:

L'objet de la formation "Les fondamentaux" est de permettre à toute personne, quel que soit son poste dans l'entreprise et son rôle dans le projet Agile, de comprendre la démarche Agile et la rupture culturelle qu'elle induit par rapport aux méthodes classiques de gestion de projets. Cette formation permet aux participants de valider un total des crédits PDUs dans le cadre de l'obtention ou du renouvellement de leur titre de **certification du PMI** (Project Management Institute), selon la répartition suivante :

Compétences techniques : 14 PDUs pour PMP®/PgMP®, 14 PDUs pour ACP®

Compétences Leadership : 0 PDU

Compétences Stratégie : 0 PDU

[En savoir plus sur l'obtention des PDUs](#) | [En savoir plus sur les titres de certification du PMI](#)

PMP, PgMP et ACP sont des marques enregistrées du Project Management Institute, Inc

Public visé:

Tous les candidats désirant s'initier aux méthodes agiles (les profils pouvant être très hétérogènes). Ce module est un prérequis pour de nombreuses autres formations - certifiantes ou non.

Objectifs:

- Comprendre cette rupture
 - Faire un choix de méthode en gestion de projet
 - Parler des méthodes agiles et de les faire partager
-

Pré-requis:

Aucun pré-requis si ce n'est l'envie de découvrir l'agilité.

Contenu:

Serious Game « Icebreaker Agile » 1

Le "Icebreaker Agile" a pour but de démarrer la journée, d'amener les participants à exprimer par leur choix ce que représentent les méthodes Agiles pour eux et partager ces représentations avec le groupe. Ils expérimentent l'auto-organisation, la limitation du temps (le timeboxing) ainsi que la possibilité d'échouer à atteindre l'objectif prévu pour une itération.

Le mouvement Agile

- Les origines d'Agile, la culture Agile, le manifeste Agile, les principes Agile
- Le cycle de vie Agile, et en quoi il diffère du cycle de vie classique
- Les différentes méthodes classiques (PMP, Prince2), leurs avantages et leurs inconvénients par rapport aux méthodes agiles
- Illustration de la supériorité des méthodes Agile sur les méthodes classiques au travers d'un cas d'utilisation spécifique

L'Agilité en action

- Panorama des méthodes agiles et de leurs apports respectifs (les 10 méthodes agiles et leurs spécificités propres)
- Présentation des deux méthodes agiles les plus répandues: SCRUM et XP; leurs similarités, leurs différences
- Présentation de méthodes connexes à Agile (Lean IT, Kanban)

Mise en Œuvre d'Agile

- Le changement dans l'entreprise, les résistances traditionnelles au changement
- Les freins au déploiement des méthodes agiles, les spécificités de leurs domaines d'application, leurs risques particuliers, les contraintes particulières à Agile
- Les mythes sur Agile, les mauvaises applications d'Agile et les facteurs clés de succès d'une implémentation réussie d'Agile

Les participants seront amenés à travailler en équipe, notamment sur des serious games (Icebreaker Agile) pour faire comprendre les valeurs Agile, leurs avantages et mettre en pratique les acquis du cours.

Autre(s) Information(s):

Un support de cours papier sera remis à chaque participant, reprenant les éléments, concepts, jeux et outils présentés lors du cours. Ce support est destiné à favoriser la mise en place par le participant des acquis de la formation au sein de leur entreprise.

Unix, Linux : Les Bases indispensables

Durée: 4 Jours

Résumé:

Cette formation présente les concepts fondamentaux d'Unix-Linux et constitue un tronc commun technique indispensable pour les informaticiens ayant à travailler sur un système Unix ou Linux quelle qu'en soit la version.

Public visé:

Cette formation s'adresse aux administrateurs et ingénieurs amenés à travailler en environnement Unix-Linux.

Objectifs:

- Acquérir les compétences indispensables pour une utilisation maîtrisée d'un système Unix ou Linux.
 - Appréhender les aspects essentiels de la programmation Shell (écriture de scripts).
 - Comprendre les fondamentaux d'Unix et sélectionner les commandes et utilitaires essentiels dans la pratique quotidienne
-

Pré-requis:

Aucune connaissance préalable sur Unix ou Linux n'est nécessaire. Avoir de bonnes connaissances générales en informatique.

Cours suivant(s):

Nous recommandons la formation UX005 Linux Administration système aux professionnels amenés à évoluer en environnement Linux.

Contenu:

Caractéristiques générales et composantes du système.

Panorama des versions.

Linux par rapport à Unix, logiciels «open source».

Connexion et déconnexion, premières commandes simples.

Accès à la documentation.

Approche utilisateur de l'organisation des disques.

Types et noms de fichiers, arborescence, commandes essentielles.

Maîtrise des droits d'accès.

Prise en main de l'éditeur de texte «vi».

Processus (définitions, attributs, commande «ps», interruptions).

Mécanismes disponibles pour l'utilisateur :

- redirections
- «pipeline»
- arrière-plan
- processus stoppés

Utilisation du shell, variables d'environnement, alias, historique des commandes, fichiers de connexion.

Manipulation de commandes essentielles :

- find
- commandes d'impression
- sauvegardes
- filtres

Manipulation de commandes réseau (telnet, «remote commands», ftp, ssh).

Comprendre les interfaces graphiques :

- principe fonctionnel
- CDE
- KDE
- GNOME

Les bases de la programmation Shell

- Procédures, paramètres, codes retour.
- Instructions (tests, boucles, aiguillage).
- Fonctions.
- Commandes internes utiles

Travaux pratiques

- Ouverture de session et premières commandes
- Nommages des fichiers et répertoires
- Manipulations de fichiers
- Droits d'accès des fichiers
- L'éditeur vi
- Gestion des process et redirections
- Éléments du Shell, variables et jokers
- Personnaliser son environnement
- Utilitaires
- Les bases de la programmation shell

Autre(s) Information(s):

Support de cours en français remis aux participants.

Gestion de projet AGILE : DEVOPS Foundation (DOI)

Durée : 2 jours

Résumé

Cette formation - accréditée par le DEVOPS Institute - permet aux participants d'acquérir les connaissances de base sur DevOps, mouvement culturel et professionnel qui met l'accent sur la communication, la collaboration, l'intégration et l'automatisation, afin d'améliorer le flux de travail entre les développeurs de logiciels et les responsables de l'exploitation IT. Des Workflows améliorés permettront d'optimiser la capacité à concevoir, développer, déployer et exploiter des logiciels et services plus rapidement.

Objectifs

A l'issue de la formation, les stagiaires seront en mesure de :

- Appréhender les objectifs et le vocabulaire DevOps
 - Découvrir les avantages de DevOps pour l'entreprise
 - Découvrir les concepts et pratiques DevOps (y compris sa relation avec Agile, Lean et IT Service Management - ITSM)
 - Découvrir les Workflows améliorés
 - Appréhender la communication et les boucles de rétroaction améliorées
 - Découvrir l'automatisation et ses dépendances
 - Appliquer DevOps dans un environnement d'entreprise
 - Comprendre les facteurs de succès et les indicateurs clés de performance
 - Aborder des exemples réels et comprendre les résultats
-

Public visé

La formation s'adresse aux consultants, informaticiens et managers impliqués dans la conception ou l'amélioration des process IT, aux fournisseurs internes et externes de services IT et aux différentes parties prenantes de process IT

Prérequis

La connaissance des services IT en général, des méthodologies Agile (Scrum) et des valeurs de l'Agilité est très fortement recommandée.

Contenu :

Introduction

- DevOps et les autres méthodes
- DevOps et l'automatisation
- Besoins et importance de l'examen

Pourquoi DevOps ?

- Perspective Business
- Perspective IT
- Les stéréotypes Dev et Ops : Perceptions et réalités

Qu'est-ce que DevOps ?

- Présentation de DevOps
- Objectifs de DevOps
- DevOps : Valeur pour l'entreprise
- Résultats démontrés
- DevOps dans l'entreprise

Les principes de DevOps

DevOps et les autres méthodes

- DevOps et Agile
- DevOps et Lean
- DevOps et la gestion des services IT (ITSM)

Culture DevOps

- Caractéristiques de la culture DevOps

Considérations organisationnelles pour DevOps

- Intervenants DevOps
- Rôles DevOps
- Les équipes DevOps

Les pratiques DevOps

- Amélioration continue
- Théorie des Contraintes
- Les 3 voies DevOps
- Agile et les méthodes Lean
- Pratiques ITSM

DevOps et l'automatisation

- Pratiques d'automatisation DevOps
- Catégories d'outils DevOps
- La chaîne d'outils DevOps

Adopter une culture DevOps

- Mise en route
- Défis, risques et facteurs clés de réussite
- Mesurer le succès de DevOps

Autre(s) Information(s)

Support de cours remis aux participants.

Programmation objet en langage Python

Durée : 3 jours

Résumé

Cette formation permet aux participants de comprendre et d'utiliser le langage de programmation Python, d'appréhender ses spécificités, ses forces et ses faiblesses. A l'issue de la formation les participants seront capables d'installer Python et un environnement de développement, quel que soit le système d'exploitation Windows, Linux, Ils seront capables de développer et maintenir des applications Python, en technologie objet, en version 2 et 3 de Python.

Objectifs

- Lire, modifier et écrire des scripts en langage Python
 - Lister, utiliser les modules Python, créer ses propres modules
 - Apprendre la programmation objet avec le langage Python
 - Comprendre les exceptions et les utiliser dans les scripts
 - Lire et écrire dans des fichiers, gérer la sérialisation des objets
 - Manipuler les bases de données relationnelles, sauvegarder ses objets
-

Public visé

Cette formation s'adresse à tout administrateur système et utilisateur averti, ayant des connaissances d'algorithmique et de programmation, ainsi qu'aux développeurs, désirant gagner en productivité et en efficacité au quotidien par le biais de programmes développés en langage Python.

Prérequis

- Connaissances de base en informatique et édition de fichiers texte.
 - Notions de programmation et d'algorithmique, idéalement avec une expérience en programmation dans d'autres langages.
 - La connaissance des concepts objet et de SQL sont un plus.
-

Contenu :

Introduction

- Présentation de Python et ses versions, historique
- Installation de Python et d'un environnement de développement (IDE)

Premiers pas

- Manipulation de l'interpréteur Python
- Manipulation de l'IDE
- Bases de la syntaxe du langage
- Aide et documentation de Python

Variables, classes et opérateurs

- Les différents types de données
- Les principales classes
- Les opérateurs

Structures de contrôle

- Structures conditionnelles
- Structures répétitives
- Autres structures

Fonctions

- Utilité et définitions
- Arguments et variables
- Appel de fonction
- Définir des fonctions

Modules et packages

- Introduction
- Importation
- Aperçu de la bibliothèque standard
- Définir des modules, des packages

Programmation Orientée Objet

- Principes des concepts objets
- Définition d'une classe, création d'objets
- Héritage et polymorphisme

Gestion des exceptions

- Présentation de la gestion des erreurs
- Structure de contrôle try, except, else, finally
- Propagation des exceptions
- Définir ses exceptions
- Déclencher des exceptions

Compléments

- Module sys, interagir avec l'interpréteur Python
- Module os, interagir avec le système d'exploitation
- Module re, les expressions régulières

Travaux pratique

Installation de python

Manipulations basiques

- Manipulation de chaînes de caractères.
- Opérations sur les listes.
- Manipulation de fichiers.

Utilisation des structures de boucle

- Itérer sur un intervalle.
- Itérer sur un dictionnaire.
- Itérer sur une liste.
- Itérer sur un fichier.

Programme complet

- Mise en œuvre de programmes en Python avec optimisation du code grâce à l'utilisation de fonctions.

Mise en œuvre d'un module et manipulation du module dans un programme en Python

Mise en pratique du Paradigme Objet sur des programmes en Python.

- Définition de classe avec méthode statique
- Mise en œuvre de la notion d'héritage

Mise en pratique de la gestion des erreurs au sein de programmes en Python

Autre(s) Information(s)

Support de cours remis aux participants. Une pédagogie progressive et active est mise en œuvre pour aborder et maîtriser les concepts avec plus de 50% de pratique. Une application de gestion sera développée au fur et à mesure des apports théoriques.

Communication orale et mon métier

Durée : 1 jour

Résumé:

Nos parents nous apprennent à parler, l'école nous apprend l'orthographe, la grammaire, la dissertation mais, nous n'apprenons pas à communiquer. Bien plus complexe qu'un simple échange de mots, nous vous proposons de découvrir ce qu'est la communication et surtout les conséquences dans le métier.

Public visé

Toute personne désirant acquérir les connaissances indispensables pour bien communiquer.

Objectifs :

- Découvrir les bases de la communication nécessaires à un dialogue constructif et professionnel.
- Connaître les outils de communication efficace
- Adapter ma communication aux exigences de mon métier

Pré-requis :

Aucun.

Contenu :

Qu'est-ce que communiquer ? Découvrir

- *Jeu* : Le téléphone arabe et ... notre mémoire ! Message clef ; seulement 10% de mémorisation de ce que nous entendons et, beaucoup de modifications et d'ajouts.
- *Jeu* : Un holdup qui n'en est pas un et ... notre imagination ! Message clef ; notre cerveau donne du sens même quand ... il n'y en a pas.
- *Jeu* : Une évaluation et ... notre référentiel ! Message clef ; même derrière des phrases de tous les jours chacun donne une mesure très différente fonction de son référentiel.
- *Jeu* : Une image à plusieurs sens et ... nos filtres ! Message clef ; chacun interprète différemment ce qu'il voit selon ses propres filtres.

Comment fonctionne la communication ? Comprendre

- *Jeu* : Une lecture surprenante et ... le fonctionnement de notre cerveau. Comprendre que chacun a construit, grâce à l'apprentissage, sa « base de données » afin de répondre aux sollicitations de la vie personnelle ou professionnelle. Cette « base de données » constitue notre référentiel et notre expérience.

- Les différentes phases de l'apprentissage
- Le VAKOG : *diagnostic*
- Les 3 canaux de la communication

Les outils pour bien communiquer- S'outiller

- La boucle de la communication et la responsabilité de la communication : *exercice de mise en œuvre*
- Les attitudes de PORTER : *exercice de mise en œuvre*
- L'implicite et l'explicite
- QQQQCP et TOAST : *exercices de mise en œuvre*
- Les questions efficaces : *exercice de mise en œuvre*

Et dans mon métier ?

- Les situations de communication de mon métier : *travail en sous-groupe*.
- Les formes de communication de mon métier : *travail en sous-groupe*.
- Le vocabulaire de mon métier : *travail en sous-groupe*.
- Mon plan d'action : *travail individuel*

Les + pédagogiques :

Pédagogie active et ludique : exercices de découverte, autodiagnostic, vidéo ...
80% pratique et 20% de théorie
PAIP-Plan d'Action Individuel de Progression

Autre(s) Information(s):

Un support de cours sera remis aux stagiaires

Les outils DevOps

Durée : 2 Jours

Résumé

Cette formation permettra de sensibiliser les stagiaires sur les différentes étapes de la mise en place d'une démarche DevOps et d'identifier les principaux outils de mise en œuvre des processus.

Objectifs

A l'issue de la formation, les stagiaires seront en mesure de :

- Présentation des principes de DevOps
- Comprendre les enjeux de la mise en place d'une meilleure collaboration entre les équipes projet.
- Comprendre les enjeux de l'automatisation de la chaîne de fabrication d'un logiciel.
- Connaître les principes, les bonnes pratiques et les outils adaptés à la mise en place des processus tels que :
 - Continuous Business Planning
 - Collaborative Development (Continuous Integration)
 - Continuous Testing
 - Continuous Release and Deployment (Continuous Delivery)
 - Continuous Monitoring
 - Continuous Customer Feedback and Optimisation
- Connaître les principales étapes de l'implémentation de DevOps dans une entreprise

Public visé

Cette formation s'adresse à tous ceux qui souhaitent comprendre les grands principes théoriques et pratiques de DevOps. Elle en donne une vision globale sans se focaliser sur une technique ou un produit spécifique.

Prérequis

La connaissance des services IT en général, des méthodologies Agile (Scrum) et des valeurs de l'Agilité est recommandée.

Contenu :

Introduction

- Historique de DevOps
- C'est Quoi DevOps aujourd'hui ?
- Pourquoi DevOps

Vue d'ensemble de DevOps

- DevOps, Lean et l'agilité
- Positionnement de DevOps dans le SI

Les processus et les bonnes pratiques

- Continuous Business Planning
- Collaborative Development (Continuous Integration)
- Continuous Testing
- Continuous Release and Deployment (Continuous Delivery)
- Continuous Monitoring
- Continuous Customer Feedback and Optimisation

Comparatifs des principaux outils adaptés aux processus dans l'écosystème DevOps

- Schéma / workflow DevOps

- Intégration Continue (Build/deploy)
- Jenkins
- Travis
- GoCD
- PHPCI
- Continuum
- Provisioning, configuration
- Ansible
- Chef
- Puppet
- Salt
- CaaS/ Container
- Kubernetes
- Docker / Docker UCP
- RKT
- Gestion de version
- Bitbucket
- Github
- Monitoring
- Nagios
- Zabbix
- Outils de test
- Selenium
- Fitnesse
- Cucumber

-
- Outils spécifiques aux technologies

Démo d'utilisation des outils DevOps.

- Démonstration des scénarii de livraison continue utilisant plusieurs des outils listés ci-dessus
- Déploiement continu d'une application Java selon différents scénarii
- Envoi et affichage des métriques d'une application ou d'un serveur sur ELK

Scénarios d'implémentation de DevOps dans une entreprise

- Les trois piliers de l'implémentation réussie :
- Les ressources humaines
- Les outils
- La méthodologie
- Construire une équipe mixte et transversale
- Fixer les objectifs

- Définir les tâches et les fonctions
- Accompagner le changement culturel
- Améliorer la communication inter-équipe
- Célébration des success stories
- Utiliser des solutions d'industrialisation des processus
- Automatiser tout ce qui peut l'être
- S'appuyer sur les logiciels libres
- Développer rapidement et piloter les évolutions
- Travailler en mode agile
- Commencer par un projet de petite taille avant de généraliser (minimiser les risques)
- Amélioration Continue par l'apprentissage
- Feed-back rapide des utilisateurs

Conclusion et perspectives

Autre(s) Information(s)

Support de cours remis aux participants.

Automatisation de la construction logicielle

Durée : 5 jours

Résumé

Cette formation vous apprendra à mettre en place l'Intégration Continue, la Livraison Continue et le Déploiement Continu du code applicatif avec Jenkins. Le but étant d'automatiser toutes les tâches manuelles et répétitives afin d'accélérer le temps de mise sur le marché de nouvelles fonctionnalités applicatives

Objectifs

L'objectif de ce cours est :

- Installer un environnement d'intégration continue basé sur Jenkins
 - Automatiser les build, l'exécution des tests et l'audit de la qualité du code source
 - Déployer sur une plateforme d'intégration continue ou de production avec Jenkins.
 - Intégrer Jenkins avec d'autres outils de l'écho système DevOps (SCM, Gestionnaire de tickets, dépôts d'artefacts...)
-

Public visé

Développeurs et chefs de projets désirant mettre en place un environnement d'intégration continue sur leurs projets..

Prérequis

Posséder Connaissances d'un langage de programmation ou de Scripting.

Contenu :

Introduction à l'intégration continue

- Définition de l'intégration continue
- Les principes de l'intégration continue
- Les prérequis de l'intégration continue
- Les différents outils de l'intégration continue
- Mise en place de l'environnement d'intégration continue

Le serveur d'intégration continue Jenkins

- Le serveur Jenkins
- Le rôle du serveur Jenkins
- Les grandes fonctionnalités
- Configuration du serveur Jenkins
- La gestion des plugins Jenkins
- Les plugins les plus utilisés
- Gestion des utilisateurs et des autorisations
- Gestion de l'espace disque
- Le monitoring du serveur Jenkins
- Création de Job Jenkins

La communication

- Indicateurs de visibilité du résultat
- Statut, Santé et tendance
- Savoir réagir
- Notification par email et flux RSS
- Communication avec des outils externes

Gestion des livrables

- Notion de dépôt d'artefacts
- Stratégie de mise à disposition du résultat construit
- Le gestionnaire de dépôt d'artefacts Archiva

L'automatisation des tests

- Les différents types de tests
- Les outils de test
- Automatisation des tests
- Les environnements de test
- Configuration des rapports de test dans Jenkins
- Consultation des rapports de test dans Jenkins

Gestion de la qualité du code

- La mise en place des métriques
- Les outils d'analyse et de reporting (Checkstyle, Findbugs, PMD, tec.)
- La génération de rapports d'analyses
- La publication des résultats dans Jenkins
- Configuration de SonarQube
- La publication des résultats dans SonarQube

Principe de la livraison continue

- Principe
- Place de la livraison dans le process DEVOPS

Configuration de la livraison continue

- Configuration dans Jenkins
- Configuration dans l'outil de gestion des livrables

Pipeline & scripting

- Création d'un pipeline de livraison
- JenkinsFile et scripting
- Automatisation des Mise en prod

Travaux pratiques

Etude de cas : Mise en place d'un scénario de migration en masse d'une architecture hébergeant plusieurs serveurs réparti dans le monde.

Autre(s) Information(s)

Support de cours remis aux participants.

Assertivité et place dans une équipe

Durée: 1 jour

Résumé:

Travailler en équipe n'est pas toujours facile et nécessite différents « réglages ». Comment rapidement et efficacement travailler ensemble ? En une journée nous vous proposons de découvrir une compétence clef nécessaire à l'efficacité collective : **l'assertivité**. « Ni hérisson, ni paillason », la juste posture pour **collaborer efficacement** avec les autres.

Public visé

Toute personne désirant améliorer sa participation collective.

Objectifs :

- Clarifier les notions de confiance en soi, estime de soi et affirmation de soi.
- Identifier le potentiel de l'assertivité dans le milieu professionnel
- S'outiller pour développer son assertivité
- Les situations assertives dans mon métier

Pré-requis :

Connaître les bases de la communication.

Contenu :

Découvrir

- L'assertivité : *questionnaire individuel sur l'estime de soi et la confiance en soi*
- Mon niveau d'assertivité : *questionnaire individuel*
- L'assertivité de mon entourage professionnel : *questionnaire individuel*
- Mon profil de conscience émotionnelle : *questionnaire individuel*

Comprendre

- Différence entre confiance en soi, estime de soi et affirmation de soi : *travail en sous-groupes*
- Les comportements types : fuite, agression et manipulation
- L'intérêt de l'assertivité pour soi, pour les autres et pour l'entreprise

- Les caractéristiques d'une personne assertive

Développer son assertivité

- Une méthode
- Les outils

S'entraîner

- Savoir faire une demande efficace : *exercice de mise en œuvre*
- Savoir mettre des limites : *exercice de mise en œuvre*
- Savoir contractualiser : *exercice de mise en œuvre*
- Savoir faire part d'un dysfonctionnement ou une critique fondée et argumentée : *exercice de mise en œuvre*
- Savoir recevoir une critique sans être déstabilisé : *exercice de mise en œuvre*

Les + pédagogiques:

Pédagogie active et ludique : des autodiagnostic, des exercices en sous-groupe, des mises en situation ...
80% pratique et 20% théorie
PAIP : Plan d'Action Individuel de Progrès

Autre(s) Information(s):

Un support de cours sera remis aux stagiaires

Plus d'informations

Pour plus de renseignements ou pour vous inscrire, merci de nous contacter au 01 78 15 34 00
info@globalknowledge.fr - www.globalknowledge.fr - 100 avenue Albert 1^{er}, 92500 RUEIL-MALMAISON

Outil de contrôle des versions GIT

Durée : 2 jours

Résumé

Cette formation permet aux participants d'apprendre à manipuler le logiciel de contrôle de version décentralisé GIT et d'acquérir toutes les notions pour utiliser GIT de manière professionnelle dans une équipe de développement informatique

Objectifs

L'objectif de cette formation est de/d' :

- Installer et configurer GIT
 - Maîtriser les commandes de base pour :
 - Gérer les dépôts distants, et savoir résoudre les conflits de version
 - Gérer les branches et les tags et fusionner les versions
 - Utiliser GitLab, GitHub
 - Gérer ses projets avec GIT
-

Public visé

Cette formation s'adresse à tous les Développeurs, DevOps

Prérequis

La connaissance du monde de développement en groupe est un plus

Contenu :

Présentation de GIT

- Historique de GIT
- Les avantages de GIT
- Principes de base : commit, tagging, ...

Installation de GIT

- Installation système
- Les paramètres de configuration (user, ignore...)
- La configuration du push et du pull
- **TP** : Installation de Git (Linux et Windows)

Les fonctionnalités

- Les commandes de base
- Création des Repository (locale et distant)
- Ajouter ou ignorer des fichiers
- Effacer, déplacer des fichiers
- Notion des Tag
- Lancer un commit, pull
- Lancer un checkout, push
- Visualiser l'historique, avancer, reculer dans l'historique
- **TP** : Créer un dépôt local et un dépôt distant dans Github
- **TP** : Manipuler les commandes de base de Git

Gestion des branches

- Présentation des branches, la branche HEAD
- Création d'une branche
- Changement de branche
- Fusion de branche
- Rebase de branche
- **TP** : Créer, manipuler et fusionner les branches

Dépôts distants

- Configuration pour accéder à un dépôt distant
- Les protocoles d'accès (ssh, https, git)
- Push, pull et fetch
- La gestion des branches distantes
- **TP** : Créer un dépôt distant avec Github
- **TP** : Le PullRequest

Les différents gestionnaires de sources

- Les Clients GIT
- Source Tree
- eGit dans Eclipse : vues (staging, history), commandes
- Les Environnements GIT
- GitHub / GitLab : présentation, Pull/Merge Requests
- **TP** : Installation d'un serveur Gitlab en local
- **TP** : Manipuler le Merge Request

Travaux pratiques

- Préparation et configuration des commandes Git
- Mise en œuvre des branches, merges et autres rebases

Autre(s) Information(s)

Support de cours remis aux participants.

Docker et Kubernetes

Durée : 7 jours

Résumé

Cette formation permet aux participants d'apprendre à manipuler un cluster Docker et conteneuriser des applications microservices grâce à des travaux pratiques qui reposent sur l'utilisation de Docker et les solutions d'orchestration Kubernetes

Objectifs

A l'issue de la formation, les stagiaires seront en mesure de :

- Introduire les nouvelles architectures logicielles à base de la technologie conteneur Docker
- Apprendre à manipuler Docker et conteneuriser des applications à travers des exemples pratiques
- Savoir configurer, approvisionner et administrer une Container-based Infrastructure par le code d'une manière automatisée.
- Acquérir les fondamentaux sur Kubernetes : installation, administration, utilisation des différents composants et déploiement des applications
- Acquérir des connaissances et des compétences de base sur les stratégies de déploiement des applications conteneurisées en production

Public visé

Cette formation s'adresse aux architectes logiciels, DevOps, administrateurs systèmes et développeurs

Prérequis

Posséder des connaissances en administration Système Linux, Réseaux, Cloud (PaaS), Intégration Continue et Livraison Continue

Contenu :

PARTIE 1 – LES FONDAMENTAUX DOCKER

Introduction

- Les différentes formes de virtualisation
- La virtualisation par conteneur
- Présentation de l'écosystème Docker
- Docker et ses concurrents
- **Travaux Pratiques** : installation de Docker

Manipulation des images

- Comment récupérer des images Docker officielles
- Docker Registry (Docker Hub et Docker Store)
- Filtrer parmi les images disponibles
- Gérer les images
- Dockerfile
- **Travaux Pratiques** : Installation des images officielles et fabrication des images personnalisées contenant instructions et configurations avancées

Manipulation des conteneurs

- Création de conteneurs
- Gestion interactive des conteneurs
- Gestion des conteneurs en tant que daemon
- Identifier l'environnement d'un conteneur
- **Travaux Pratiques** : Création d'un conteneur basé sur une image prédéfinie et le mettre à jour

Les conteneurs

- Docker et les réseaux
- Le port forwarding (PAT)
- Liaisonner des conteneurs
- Utiliser les volumes pour persister des données ou les mutualiser
- Multi-conteneurs avec Docker Compose
- **Travaux Pratiques** : Mise en place d'un portail WordPress à partir de 2 conteneurs : Mysql et WordPress
- **Travaux Pratiques** : Mise en place d'une application Web MEAN Stack à partir de 2 conteneurs : NodeJS et MongoDB

PARTIE 2 – ORCHESTRATION DES CONTENEURS

Orchestration des conteneurs : Présentation et enjeux

- Cluster Docker vs Cluster Classique (Virtualisation)
- Solutions d'orchestration open-source et leurs différences
- Container as a Service (CaaS) dans le Cloud
- Docker Data Center
- Docker Machine

Docker Swarm

- Présentation et Architecture de Swarm

- Manipulation des Stacks, services et réplication
- Scaling
- Administration d'un cluster Swarm avec Portainer Web Console.
- **Travaux Pratiques** : Création d'une Container-based Infrastructure avec Docker Machine (Déploiement en local avec le driver Virtual Box, sur un Cloud Privé Openstack et sur un Cloud Public Amazon AWS).
- **Travaux Pratiques** : Mise en place d'un cluster Swarm avec Docker Machine

Kubernetes

- Présentation et Architecture de k8s
- Composants clés (ETCD, Scheduler, Serveur API, Controller Manager, Kubelet)
- Gestion des réseaux (kubenet, CNI providers)
- Modes de déploiement (All in One et Multi-node)
- Outils d'administration (Kubectl, Remote CLI)
- Les ressources (Pods, Namespaces, Services, Deployments, Replication Manager, ConfigMaps, Secrets, etc)

Travaux pratiques

Introduction

- installation de Docker

Manipulation des images

- Installation des images officielles et fabrication des images personnalisées contenant instructions et configurations avancées

Manipulation des conteneurs

- Création d'un conteneur basé sur une image prédéfinie et le mettre à jour

Les conteneurs

- Mise en place d'un portail WordPress à partir de 2 conteneurs : Mysql et WordPress
- Mise en place d'une application Web MEAN Stack à partir de 2 conteneurs : NodeJS et MongoDB

Docker Swarm

- Gestion des volumes (Volumes Persistant, Claims, Volumes Providers)
- Sondes Liveness et Readiness
- Scaling et Auto-scaling
- Exposition publique et Routing des applications (Ingress, Reverse-Proxy).
- **Travaux Pratiques** : Mise en place de Kubernetes avec Vagrant et Ansible (1 Master et 3 Nodes)
- **Travaux Pratiques** : Administration Kubernetes (Kubectl)
- **Travaux Pratiques** : Pods, Labels, Selectors et Namespaces
- **Travaux Pratiques** : Replica, Deployments et Services
- **Travaux Pratiques** : Gestion des volumes
- **Travaux Pratiques** : Déploiement d'une application stateful
- **Travaux Pratiques** : LoadBalancer avec Ingress
- **Travaux Pratiques** : Strategies de déploiements avancées (Rolling update, Blue/Green et A/B Testing)

- Création d'une Container-based Infrastructure avec Docker Machine (Déploiement en local avec le driver Virtual Box, sur un Cloud Privé Openstack et sur un Cloud Public Amazon AWS).
- Mise en place d'un cluster Swarm avec Docker Machine

Kubernetes

- Mise en place de Kubernetes avec Vagrant et Ansible (1 Master et 3 Nodes)
- Administration Kubernetes (Kubectl)
- Pods, Labels, Selectors et Namespaces
- Replica, Deployments et Services
- Gestion des volumes
- Déploiement d'une application stateful
- LoadBalancer avec Ingress
- Strategies de déploiements avancées (Rolling update, Blue/Green et A/B Testing)

Autre(s) Information(s)

Support de cours remis aux participants.

Développer une attitude « orientation CLIENT »

Durée: 1 jour

Résumé:

L'explosion des chartes qualité et des contrats de service type SLA (Service Level Agreement) traduisent une forte attente de qualité des clients. L'orientation client est avant tout un état d'esprit, une posture qui s'acquière. Être force de proposition au regard du besoin du client tout en mobilisant toutes les parties prenantes, tant internes qu'externes. Un vrai challenge.

Public visé

Toute personne désirant adopter une orientation client interne ou externe.

Objectifs :

- Comprendre son rôle de prestataire de service interne ou externe.
 - Développer un état d'esprit et des attitudes de service.
 - Mettre en œuvre l'orientation client dans sa façon de traiter les demandes.
 - Faire face positivement aux suggestions et manifestations d'insatisfaction des clients internes ou externes.
 - Trouver comment faire mieux pour ses clients internes ou externes.
-

Pré-requis :

Les bases de la communication.

Contenu :

Découvrir

- Clarifier ses représentations personnelles du client.
- Mieux se connaître pour mieux comprendre ses réactions face au client.
- Dépasser ses limites personnelles.
- Être conscient de son rôle.
- Choisir clairement son attitude.
- Éviter les erreurs fatales.
- Connaître les sources d'insatisfaction et de satisfaction d'un client interne ou externe.
- Distinguer attentes opérationnelles et relationnelles.

Comprendre

- Écouter, entendre et ressentir les besoins du client interne ou externe.
- Comprendre et analyser la charte qualité ou le contrat de service.
- Définir "être orienté client".
- Identifier la performance et l'objectif de satisfaction de nos clients.

- Situer son rôle et ses missions.
- Respecter et considérer son client.
- Développer la relation interpersonnelle.
- Redéfinir la relation client/ fournisseur.
- Analyser les attentes du client, y répondre et être force de proposition.

S'outiller et s'entraîner

- Respecter la personne et ses besoins versus l'entreprise et ses besoins.
 - Clarifier et fiabiliser
 - Adopter une attitude positive face au mécontentement.
 - Développer l'esprit de service dans son métier.
 - Avoir le souci de réussir sa mission
 - Mettre en œuvre une charte dans son comportement quotidien.
 - Apporter des solutions appropriées et au-delà
-

Les + pédagogiques:

Formation spécifiquement dédiée à des équipes informatiques. Pédagogie active : mises en situation, jeux de rôles, échanges, quiz... et sur des exemples de SLA, charte ou contrat de service. 80% pratique et 20% théorie
PAID : Plan d'Action Individuel de Développement

Autre(s) Information(s):

Un support de cours sera remis aux stagiaires

Infrastructure par le Code : Puppet, Ansible, Terraform, Vagrant

Durée : 5 jours

Résumé

Cette formation permet aux participants d'apprendre les outils infrastructure par le code et de comprendre ce qu'est une infrastructure codée et pourquoi c'est un point important du DevOps.

Objectifs

L'objectif de cette formation est de/d' :

- Appréhender l'organisation d'Ansible et de Puppet
- Savoir utiliser et créer des playbooks et des modules pour gérer une infrastructure.
- Coder une infrastructure avec Terraform et Vagrant.
- Appréhender les différents outils Infrastructure par le code avec leurs spécificités

Public visé

Cette formation s'adresse aux DevOps, Développeur, Architecte, Administrateur

Prérequis

Posséder des connaissances d'administration Linux et de langages DevOps

Contenu :

PARTIE 1 – OUTILS DE Création et gestion des environnements virtuels par le code

Terraform

- Retour sur les types d'infrastructure
- Terraform : présentation, architectures et fonctionnalités
- Gestion de multi-providers (Cloud, Hyperviseur)
- Gestion d'une infrastructure : création, modification
- Utiliser l'interface en lignes de commande (CLI)

Vagrant

- Installation de vagrant
- Gestion des plug-ins et des boxes (VirtualBox, Cloud)
- VagrantFile
- Les commandes de base
- Cycle de vie d'une instance Vagrant
- Gestion du réseau
- Synchronisation des fichiers

CONFIGURATIONS

Présentation Puppet

- Pourquoi la gestion d'infrastructure centralisée ?
- Pourquoi Puppet ?
- Les différentes versions et distributions de Puppet.
- Le fonctionnement général de Puppet.

Mise en œuvre

- Installer un Puppet master.
- Configurer un Puppet master.
- Installer et configurer l'agent Puppet.

Prise en main du langage de Puppet

- Les principaux types supportés par Puppet.
- La syntaxe du langage.
- Générer des fichiers à partir de templates.
- Les relations entre ressources.
- Les ressources définies.
- La remontée d'informations avec Facter.

Réutilisation de modules Puppet

PARTIE 2 – OUTILS DE GESTION DES

- Pourquoi réutiliser des modules existants ?
- Les modules, les classes.
- Utiliser un module existant.
- Pourquoi structurer son code Puppet sous forme de modules réutilisables ?
- Ecrire un nouveau module.

Bonnes pratiques pour déployer et utiliser Puppet

- Le formatage du code.
- Utiliser la gestion de versions et l'intégrer avec Puppet.
- Séparer le code générique des paramètres spécifiques à un système

Présentation Ansible

- Présentation et énumération des thèmes du cours.
- Introduction à Ansible

Travaux pratiques

- Mise en place de Terraform
- Création d'une instance Amazon avec Terraform
- Installation et Configuration de Vagrant
- Création d'un environnement virtuel avec Vagrant et VirtualBox
- Installation des Plugins et Gestion de réseau
- Installation d'une infrastructure de base Puppet (Master Puppet, Serveur DNS)
- Les ressources (File, User, Cron, Exec)
- Les variables, Expressions et Facts
- Déploiement basique et Github

- Décrire la terminologie et l'architecture d'Ansible.
- Déploiement d'Ansible
- Installer Ansible et exécuter des commandes ad hoc.
- Mise en œuvre de playbooks
- Gestion des variables et des inclusions
- Décrire la portée et la priorité des variables,
- Mise en œuvre du contrôle des tâches
- Mise en œuvre de rôles
- Créer et gérer des rôles

Ansible Tower

- Mise en œuvre d'Ansible Tower
- Appréhender l'interface graphique Reporting.

- Déploiement d'apache2 avec Virtualhosts
- Déploiement de Wordpress
- Déploiement d'un service DHCP
- Mise en place d'une infrastructure de base Ansible
- Connexion et Exécution d'un script simple
- Mon premier PlayBook
- Rôles, Fact et Idempotence
- Un exemple plus complet de configuration de serveur web
- Installation et Configuration d'Ansible Tower avec Vagrant
- Ansible Tower

Autre(s) Information(s)

Support de cours remis aux participants.

Métrologie, Monitoring et Gestion des Logs

Durée : 5 jours

Résumé

Cette formation permet aux participants d'apprendre à mettre en place une pille logicielle de monitoring, de collection des métriques et de gestion d'alertes avec la stack Prometheus et Grafana. Cette formation va permettre aux participants de configurer et administrer un cluster ELK pour gérer efficacement les logs applicatifs

Objectifs

L'objectif de ce cours est :

- Savoir installer, configurer et administrer une stack de supervision avec Prometheus et Grafana
 - Connaître et comprendre le fonctionnement de Prometheus et Grafana
 - Être en mesure de mettre en œuvre la supervision dans un environnement hétérogène (Cloud)
 - Identifier les bonnes pratiques à mettre en place pour développer une application basée sur la stack ELK
 - Découvrir les bases de la gestion de messages avec Logstash et le stockage de données massif avec Elasticsearch
 - Construire des graphes représentatifs des logs avec Kibana
-

Public visé

Cette formation s'adresse aux Administrateur Systèmes, Ops, Architecte

Prérequis

Posséder des connaissances en Administration Systèmes, Linux, Réseaux, HTTP

Contenu :

PARTIE 1 – Métriques, Alertes et Monitoring

Introduction

- Différents type de supervision (Infrastructures, Services et Applications)
- Solutions disponible sur le marché
- Définition d'une stratégie de monitoring
- Pourquoi Prometheus ?

Administration Prometheus

- Prometheus : fonctionnalités, principe de fonctionnement
- Prometheus vs autres outils de supervision
- Architecture de Prometheus (Short Jobs, Job Exporter, PushGateway, Service Discovery, Kubernetes)
- Stockage : présentation des données à stocker, (types/formats), stockage par double delta
- Prometheus server : installation et configuration
- Langage PromQL
- Jobs exporter : installation, configuration
- Définition d'alertes

Administration Grafana

- Faiblesses de l'interface de Prometheus
- Grafana : présentation de l'interface web
- Grafana : Sources de données et Dashboard disponibles

PARTIE 2 – Gestion des Logs

Introduction à la Stack ELK

- Elasticsearch, Logstash et Kibana
- Cas d'utilisation

Travaux pratique

Administration Prometheus

- Installation et Configuration de Prometheus
- Monitorer un nœud avec Node Exporter
- Monitorer un serveur web avec Apache Exporter
- Monitorer une base de données avec MySQL Exporter

Administration Grafana

- Branchement et Interfaçage avec Prometheus
- Configuration d'un dashboard
- Configuration du gestionnaire d'alertes

- Représentation des données dans Elasticsearch
- Architecture générale et Ecosystème Elastic
- Elasticsearch : une base de donnée ou un moteur de recherche
- Installation et configuration de base
- Communication avec avec Elasticsearch
- Structure de l'API
- Stockage : présentation des données à stocker, (types/formats), Query

Récupération des Logs avec Logstash

- Fonctionnement et concepts
- Positionnement des Beats par rapport à Logstash
- Installation et configuration de base
- Filters, Inputs /Outputs, Index

Stockage des logs dans Elasticsearch

- Installation, configuration de base et plugins
- Le rôle et l'importance du mapping
- Recherche basique
- Notion d'Agrégats

Visualisation des logs avec Kibana

- Le rôle de Kibana dans la stack ELK
- Installation, configuration de base et plugins
- Rechercher, agréger, visualiser, sauver, exporter

Introduction à la Stack ELK

- Mise en place de la Stack ELK

Récupération des Logs avec Logstash

- Mise en place d'une stack ELK
- Récupérer des logs et métriques avec Logstash

Stockage des logs dans Elasticsearch

- Stocker et rechercher les logs avec Elasticsearch

Visualisation des logs avec Kibana

- Construire des graphes représentatifs avec Kibana

Autre(s) Information(s)

Support de cours remis aux participants.

Adopter la posture attendue en milieu professionnel

Durée: 1 jour

Résumé:

Au-delà des règles de politesse, l'entreprise possède ses codes qu'il convient de connaître pour faciliter son intégration. Cette journée vous donne les clefs pour identifier, avant et pendant votre intégration, les règles à respecter ainsi que rappeler les usages en entreprise.

Public visé

Toute personne intégrant une nouvelle entreprise.

Objectifs :

- Comprendre les enjeux de la posture professionnelle
 - Identifier les spécificités de l'entreprise
 - Connaître son profil dans une équipe
 - Adapter ma posture aux différents interlocuteurs de l'entreprise
-

Pré-requis :

Aucun.

Contenu :

Découvrir

- L'entreprise :
 - 2 vidéos pour commencer : *débriefing collectif après visionnage*
 - Un vocabulaire à connaître : *exercice en sous-groupes*
- Moi :
 - Quel rôle dans une équipe ? *Diagnostic individuel*
 - Quelles petites voix intérieures ? *Diagnostic individuel*

Comprendre

- L'entreprise :
 - Ses différents acteurs : *exercice en sous-groupes*
 - Sa culture : *exercice en sous-groupes*
 - Les relations professionnelles : *exercice en sous-groupes*
- Les salariés :

- Chacun un rôle dans une équipe
- Chacun ses petites voix intérieures
- Chacun ses besoins relationnels
- Chacun sa conscience et son inconscience

Se préparer

- L'entreprise et les salariés :
 - Selon les niveaux de Dilts
 - Savoir être et compétence professionnelle : *mise en situation*
 - Les erreurs à éviter : *mise en situation*
 - Une relation responsable : *mise en situation*
 - La prise de contact : *mise en situation*
 - Les environnements de travail : *travail en sous-groupes*
 - Sa mission : *mise en situation*
 - La carte des responsabilités : *travail individuel*
-

Les + pédagogiques:

80% de pratique et 20% de théorie

Pédagogie active et ludique : vidéos, exercices de découverte, autodiagnostic, travail en sous-groupe....

PAIP : Plan d'Action Individuel de Progrès

Autre(s) Information(s):

Un support de cours sera remis aux stagiaires

Savoir se vendre en entretien

Durée: 2 jours

Résumé:

Une fois l'entretien décroché un vrai challenge s'annonce : réussir son entretien d'embauche ! Grâce à cette formation vous découvrirez comment valoriser vos compétences en valeur ajoutée pour le client, comment utiliser votre gestuelle pour dégager assurance et aisance relationnelle, comment rebondir en entretien, poser des questions et passer les différentes étapes clefs de cet entretien.

Public visé : Toute personne désirant améliorer l'efficacité de son entretien d'embauche.

Objectifs :

- Comprendre les enjeux de l'entretien pour le recruteur et pour vous
- Clarifier l'objectif de l'entretien d'embauche
- Valoriser vos compétences et votre expérience
- Préparer et réussir son entretien
- Être à l'aise dans les échanges avec le recruteur
- Savoir clore l'entretien et connaître les prochaines étapes

Pré-requis : Aucun.

Contenu :

Notre 1^{er} outil : la communication verbale et non verbale ; rappels

- La boucle de la communication / Les canaux de la communication / Le VAKOG / Les attitudes de PORTER / Les questions efficaces : *questionnaire individuel*

Quel communicant suis-je ?

- Mon vocabulaire / Ma voix / Ma gestuelle et mon regard / Mon profil de communicant : *Diagnostic individuel*

Quels enjeux ?

- Vidéo

Où en suis-je ?

- Questionnaire individuel

En amont de l'entretien : Tout préparer

- Les bonnes informations à aller chercher : *exercice en sous-groupes*
- Au-delà de mon CV, la valorisation de mes compétences et expérience : *exercice en sous-groupes*
- Ma valeur ajoutée sur le poste : *exercice individuel*

- Préparation minutieuse des différentes étapes de l'entretien : *exercice en sous-groupes*
- Préparation psychologique de l'entretien : *mise en situation*

Pendant l'entretien : Donner envie au recruteur

- L'arrivée dans l'entreprise : *exercice en sous-groupes*
- Le 1^{er} contact avec le recruteur : *mise en situation*
- Rester vrai et sincère : *mise en situation*
- Maîtriser l'échange tout au long de l'entretien : être totalement présent : *mise en situation*
- Prendre congé efficacement : *mise en situation*

Après l'entretien : Rester mobilisé

- Les remerciements : *exercice en sous-groupes puis mise en situation*
- Les relances : *exercice en sous-groupes puis mise en situation*
- Les réseaux sociaux : *exercice en sous-groupes*

Exercices

- Vidéo-Journal TV de 20H : Etude de vidéos
- Les étapes de l'entretien
- L'e-réputation
- Présenter son métier et ses compétences
- Rédaction d'un CV
- Lettre de motivation
- Préparation de l'entretien
- Jour J ; Déroulement de l'entretien : Arrivée, présentation, fin de l'entretien

Les + pédagogiques:

Pédagogie active et ludique : vidéos TOP/FLOP, entraînements aux entretiens filmés, exercices individuels et collectifs
80% pratique et 20% théorie
PAID : Plan d'Action Individuel de Développement