Formation Microservices : optimisez vos architectures

Concevoir, développer, tester, sécuriser et gérer des applications compatibles avec l'architecture microservices.

N° ACTIVITÉ : 11 92 18558 92 | TÉLÉPHONE : 01 85 77 07 07 | E-MAIL : inscription@hubformation.com

Microservices, Web Services, Serverless, Rest, API, Web API, API Management et Service Mesh, Cloud Native Applications, DevOps, Event Storming, Domain Driven Design, Orchestration, Conteneurisation, etc. Les Architectures de Microservices constituent des pratiques modernes dans l'industrie de fabrication de logiciels. Elles apportent des innovations pour fabriquer des logiciels de plus en plus performants, nativement scalables, élastiques et sécurisés "by design".
Beaucoup de pratiques sont aujourd'hui éprouvées et matures pour être répliquées de manière industrielle dans les entreprises. Mais des faiblesses, voire des lacunes, persistent et doivent être comprises et prises en compte pour éviter la désillusion.
Cette formation aborde le sujet des Microservices de manière très cohérente sur une dizaine de dimensions : définition, enjeux business, défis technologiques, architectures techniques, usages et UX, outils du marché, méthodes, sécurité et cybersécurité, culture d'entreprise et gouvernance.
A l'issue de ces deux journées, vous aurez acquis les aptitudes nécessaires pour comprendre les Architectures de Microservices ainsi que les technologies et concepts qui gravitent autour : Web Services, API, Serverless, DevOps, etc. Les sujets abordés vous donneront une vision cohérente et globale, mais aussi techniquement approfondie sur les Architectures de Microservices.
Les connaissances acquises sont toutes pratiques et immédiatement actionnables pour améliorer les conditions de succès de vos projets de construction logiciels modernes.

Référence	WES32
Durée	2 jours (14h)
Tarif	2 197 €HT
Repas	repas inclus

SESSIONS PROGRAMMÉES

PARIS

du 29 au 30 septembre 2025
du 1er au 2 décembre 2025

VOIR TOUTES LES DATES

Objectifs

| Identifier l'architecture technique et fonctionnelle des Web Services, Microservices et API.
| Concevoir, organiser et délivrer un SI digital fondé sur les Microservices et les APIs : architecture technique, méthode de conception, choix de langages de développement, déploiement des pratiques d'administration et d'exploitation, ouverture des API aux partenaires, maîtrise de la sécurité,...
| Analyser les secrets des géants du Web qui conçoivent et développent des centaines de Microservices par mois et les ouvrent par API aux clients et aux développeurs, tout en maîtrisant les risques.

Public

| décideurs informatiques
| urbanistes et architectes
| consultants
| ingénieurs
| chefs de projets, maîtrises d'ouvrage.

Prérequis

| Connaissances de base en systèmes d'information.

Programme de la formation

Introduction aux concepts
Les Microservices
| Définition formelle et non interprétable de la notion de Microservice.
| Quelle différence entre un Web Service et un Microservice ?
| Critères formels d'éligibilité au Microservice.
| Qu'est-ce que la granularité d'un Microservice ?
| Qu'est-ce que l'empreinte technique d'un Microservice ?
| Spécificités Cloud pour un Microservice.

Webservices
| Définition de la notion de Service en architecture technique.
| Quelle différence entre un Web Service et un composant ?
| Quelles différences entre un Web Service et une application Web ?
| Quelle place pour les Web Services en interne à l'entreprise.
| Les surcoûts spécifiques aux Web Services.
| Les limites de la prolifération des Web Services dans l'entreprise.

API
| Définition formelle et non interprétable d'une API.
| Quel rapport entre Microservice et API ?
| Définition de l'écosystème d'une API.
| Définition de API Gateway.
| Qu'est-ce que l'API Management ?
| Introduction aux architectures Service mesh.

Microservices et Serverless
Architectures de Microservices
| Les 4 principaux modèles d'Architectures Microservices.
| Introduction aux principes de l'architecture événementielle.
| Introduction au pattern d'architectures Event Driven.
| Quelles sont les spécificités des approches d'architectures Message Driven ?
| Les principaux pièges dans les pratiques de Command Sourcing.
| Domaine d'application du pattern CQRS (Command, Query Responsibility Segregation).
| Les principaux pièges dans le choix du pattern d'Architecture Microservice et comment les éviter.

Architectures Serverless
| Introduction au concept Serverless.
| Les domaines d'application privilégiés pour le Serverless.
| Serverless et Microservices : les principaux pièges et comment les éviter.
| Les pré-requis pour tirer profit des architectures Serverless.
| La problématique de la portabilité entre fournisseurs de Serverless Backends.
| Les précautions essentielles pour assurer la portabilité d'une architecture serverless.

Données et Microservices
Transactionnel en Microservices
| Rappel des principes transactionnels ACID.
| Principes des transactions de compensation fonctionnelle.
| Introduction à la conjecture CAP (Consistency, Availability, Partitioning).
| Comment gérer une transaction longue entre plusieurs Microservices ?
| Exemples de réalisations opérationnelles grâce au théorème CAP.
| Exemples de reprise fonctionnelle d'une transaction.
| Étude de cas d'une AMS robuste "By Design".

Données et Microservices
| Quels types de relations entre bases de données et Microservices ?
| Comment délimiter la responsabilité technique des données ?
| Les bons et mauvais arguments pour partager les données entre microservices.
| Introduction aux bases de données NoSQL.
| Les 4 principaux types de bases de données NoSQL.
| Étude de cas d'une architecture avec 5 types de bases de données.

Invocation et composition de Microservices
Standards Techniques des Microservices
| HTTP : rappel des principes du protocole.

| Quel intérêt à utiliser HTTP comme un protocole applicatif ?
| HTTP et les microservices : intérêt et alternatives.
| JSON : rappel des principes.
| Les limites de JSON en langage de configuration.
| Que penser des langages alternatifs : TOML, YAML, HOCON, HJSON, etc. ?

Patterns d'invocation de Microservices
| Rappel des principes d'invocation RPC (Remote Procedure Call)
| Avantages et inconvénients de l'Invocation REST HYPERMEDIA.
| Comment invoque un microservice en mode Message ?
| Introduction aux techniques d'invocation réactive de Microservice.
| Rappel des principes fondamentaux du style d'architecture REST.
| Comprendre la rupture profonde apportée par l'Hypermédia.
| GraphQL : les principaux pièges et comment les éviter.

Workflow, Orchestration et Chorégraphie
| Les 3 approches techniques pour coordonner les appels de Microservices.
| Workflow, Orchestration et Choregraphie : avantages et inconvénients respectifs.
| Comment la conception fonctionnelle influence l'option WOC ?
| Les principaux langages WOC et comment choisir ?
| Étude de cas de coordination d'appels entre 10 Microservices.
| Etude de cas : orchestration "By Design" ?

Conception de Microservices
| Démarche de conception d'une application à base de Microservices.
| Introduction aux concepts de "protocole métier" et "espace sémantique"..
| Introduction aux pratiques de Checkpoints.
| Que penser de l'approche fondée sur le "Event Storming" ?
| Les difficultés à bien connaître pour réussir une session "Event Storming" ?
| Comment définir les indicateurs de mesure qualitative et quantitative ?
| Études de cas qui illustrent le réalisme et l'enlisement de la phase conception.

Domain Driven Design (DDD)
| Définition du concept DDD (Domain Driven Design)
| Les points forts des Aggregates.
| Comment bien délimiter les Bounded Context ?
| Introduction à la notion de "'Ubiquitous Language".
| Les points sensibles lors du Mapping entre Bounded Context et Microservices.
| Que penser du Principe de l'Unité de Cohésion dans DDD.
| Domaine de sens vs Domaine d'intérêt : intérêts et limites.

UX et Microservices
| Quel impact des microservices sur l'UX et l'UI.
| Etude de cas d'une UI dynamique.
| Que penser de l'approche de Micro UI ?
| Les pratiques modernes de construction fonctionnelle l'UI par les microservices ?
| Intérêt à générer l'UI dynamiquement.
| Etude de cas du cas UX/UI et Microservices de Netflix, Google Maps et Amazon.

Cycle de vie des Microservices
Développement de Microservices
| Comment organiser les équipes de développement ?
| Comment distribuer les responsabilités des développeurs sur les microservices ?
| Comment gérer la scalabilité des équipes de développement ?
| Les principaux ratios entre profils expert par microservice.
| Pourquoi le langage de développement est mineure dans une architecture de microservices ?
| Introduction au CI (Continuous Integration) et CD (Continuous deployment).
| Les limites d'une plateforme CI (Continuous Integration) et CD (Continuous deployment) et comment les repousser.

Tests de Microservices
| Comment tester des centaines de microservices tous indépendants ?
| Comment faire évoluer et gérer les versions des Microservices ?
| Introduction aux tests de Chaos ?
| Les idées reçues sur les tests de performance et de montée en charge ?

| Les précautions essentielles pour réussir les tests de sécurité et cybersécurité ?
| Introduction aux architectures de Microservices "Secured By Design", "Scalable By Design".

Déploiement, Exploitation et DevOps
| Plateforme DevOps vs Pratiques DevOps.
| Quelle place pour le IaaS, IaC, CaaS, PaaS dans une plateforme Microservices ?
| Comment faire de la mise en production devient un non-évènement ?
| Comment faire du crash d'un microservice un non-évènement ?
| Les nouveaux défis dans l'administration des données des Microservices.
| Comment industrialiser le déploiement des bases de données des microservices ?
| Comment administrer des milliers d'instances de microservices ?
| Introduction aux problèmes de l'Observabilité en mode Runtime.

Sécurité des microservices
| Les principes de base en Design d'architectures de microservices robustes.
| Comment simplifier et uniformiser la mise en oeuvre de la sécurité ?
| Étude de cas avec oAuth et OpenID.
| Comment contrôler finement les autorisations ?
| Comment concilier ouvertures d'API et sécurité.
| Introduction à la sécurité applicative.
| Introduction aux techniques de durcissement fonctionnel.

Gouvernance des Microservices
Défis et faiblesse des Microservices
| A quelle vitesse durable opérer la transformation de l'existant ?
| Quels sont les risques post rupture culturelle ?
| Comment gérer culturellement l'éclatement des structures de données ?
| Comment faire face à la complexité multidimensionnelle ?
| Que faire des salariés qui ne peuvent pas supporter la rupture technologique.
| Comment opérer la rupture dans la culture DevOps ?
| Quel discours face aux exigences d'hyper automatisation et industrialisation ?

Gouvernance
| Modèles d'organisation autour d'une plateforme microservices.
| Les limites définitives de l'organisation MOA/MOE.
| La problématique des organisations bi-modales.
| Mode Projet vs mode Produit
| Comment allouer les budgets en approche Produit ?
| Comment gérer l'échec "Fail Fast" dans une grande entreprise ?
| Introduction à la loi Werner Vogels : "You build it, you run it".
| Comment tirer le meilleur parti des frameworks agiles (Scrum, Kanban, SAFe, etc.) ?
| Les points sensibles pour catalyser une transformation culturelle.
| Réflexions contemporaines sur l'Humain face aux architectures complexes et automatisées.

Panorama
Ce chapitre dresse un panorama critique de l'offre du marché, des outils et des plateformes pour concevoir, développer, déployer, exploiter, administrer et sécuriser des architectures de Microservices, Web Services et API : typologie des acteurs et de l'offre de standards, de normes et de produits, caractéristiques des principaux produits dans chaque catégorie, évolutions fonctionnelles et techniques attendues, points forts et limites, domaines d'application privilégiés et principales contre-indications respectives, etc.

Méthode pédagogique

Chaque participant travaille sur un poste informatique qui lui est dédié. Un support de cours lui est remis soit en début soit en fin de cours. La théorie est complétée par des cas pratiques ou exercices corrigés et discutés avec le formateur. Le formateur projette une présentation pour animer la formation et reste disponible pour répondre à toutes les questions.

Méthode d'évaluation

Tout au long de la formation, les exercices et mises en situation permettent de valider et contrôler les acquis du stagiaire. En fin de formation, le stagiaire complète un QCM d'auto-évaluation.

Accessibilité

Les sessions de formation se déroulent sur des sites différents selon les villes ou les dates, merci de nous contacter pour vérifier l'accessibilité aux personnes à mobilité réduite.
Pour tout besoin spécifique (vue, audition...), veuillez nous contacter au 01 85 77 07 07.