



Le Campus Numérique in the Alps

Formation DevOps

Objectifs de la formation

La formation DevOps est un parcours de 1729 heures formant au métier de DevOps. Le DevOps a un rôle essentiel dans les processus de développement actuels en entreprise, il est réellement le pivot entre le rôle de développeur et l'administrateur de projets de développement. Ses casquettes sont multiples, il évolue dans un contexte Agile et fluidifie les échanges entre les protagonistes d'un projet. Il est bon communicant.

Il dispose d'une très bonne connaissance de la stack technique de développement, il est full-stack. A savoir, il maîtrise autant les couches basses (back-end) que haute (front-end) du développement d'un site ou une application. Il maîtrise les mécanismes et la pyramide des tests ainsi que les notions de qualité de code et est capable de prodiguer des recommandations éclairées aux équipes.

Il a également en charge la mise en place et l'administration des outils assurant la pérennité et la fluidité du développement (conteneurisation, gestionnaire de tickets, outils d'analyse automatique de qualité de code, déploiement automatique, tests automatiques).

Au terme de cette formation, vous serez en mesure de gérer des projets informatiques avec une très bonne maîtrise des notions de qualité de code, des processus Agiles, des méthodes de collaboration entre équipes de développeurs. Cette formation s'accompagne également de tous les outils permettant de favoriser les processus de relation client dans le développement d'une application, et ce, à l'aide des méthodologies de gestion de projet Agile, des mécanismes de livraison continue et de gestion et traitement des retours utilisateurs lors des différentes phases de développement. Ce cursus adresse enfin la sécurisation des applications en étudiant les principales failles de sécurité présentes dans les projets actuels.

En un mot, cette formation permet de développer en équipe de code pérenne, robuste, de qualité, adapté aux besoins des utilisateurs finaux. Un fort accent est mis sur les *soft-skills* afin de fluidifier au maximum les processus de développement et les relations clients / équipes de développement.

Ce programme de formation est cumulatif avec un programme de développeur concepteur et développeur d'applications, il permet d'aller plus loin dans la spécialisation.

Les activités principales du concepteur développeur d'applications sont :

- **Analyse et formalisation des besoins du client** : définition des spécifications fonctionnelles de l'application logicielle en vue de son développement (étude technique des besoins exprimés par les utilisateurs, analyse des étapes de fonctionnement, spécification de la solution technique envisagée) ;

- **Conception et modélisation des applications informatiques** : modélisation et définition du périmètre de l'Interface Homme Machine dans le respect du cahier des charges et dans la réalisation des maquettes des interfaces ;
- **Développement des applications** : évaluation et définition des solutions techniques les plus adaptées (programmation, tests, validation des fonctionnalités développées) ;
- **Déploiement sur site et formation des utilisateurs** : conception des notices techniques d'installation, des guides d'utilisateur et d'administration, des formations et leurs supports ;
- **Maintien, correction et évolution des applications** : définition des évolutions et correctifs nécessaires à la continuité de l'application, et information des utilisateurs.

Compétences développées

Le programme aborde les compétences attendues sur les technologies majeures : PHP, HTML, JavaScript, Java, Node JS, CSS, Angular, Symfony, UX UI, bases de données et administration des réseaux, algorithmique... et les soft skills indispensables en entreprise. Les méthodes d'apprentissage « apprendre à apprendre » permettent aux apprenants de se former en autonomie tout au long de leur carrière pour suivre les évolutions technologiques.

Compétences développées à l'issue de la formation :

- Traduire techniquement un cahier des charges
- Mettre en œuvre une méthodologie pour définir les fonctionnalités d'une application
- Concevoir les fonctions et les algorithmes associés et spécifier les flux de données
- Etablir le schéma relationnel de la base de données et la créer
- Maquetter les interfaces et les créer
- Ecrire des requêtes sur la base
- Utiliser un langage de programmation pour coder les algorithmes, les interfaces d'entrées-sorties réactives dans le cadre de systèmes industriels et mobiles
- Définir des jeux de tests, tester et déboguer l'application créée
- Etablir les documents à l'attention des différents utilisateurs (entreprise, équipe projet, usagers)

Le programme permet également de développer les compétences Dev Ops attendues par les entreprises dans :

- La mise en application de la méthodologie de gestion de projet Agile
- Le développement informatique
- Le processus de déploiement et de livraison continue
- La conteneurisation et le déploiement des applications
- L'analyse de la qualité du code
- La sécurisation des applications.

Méthodes pédagogiques

Le Campus adopte une approche par compétences, les modules de formation étant construits sur la base de compétences à acquérir : savoirs, savoir-faire, savoir-être, outils, méthodologies... Cette approche permet d'assurer une formation professionnalisante avec le développement de compétences opérationnelles correspondant aux exigences des métiers visés.

Le modèle de pédagogie par projet mis en place dans les modules de formation permet de rendre les stagiaires acteurs de la formation et de l'apprentissage. L'autonomie développée dans les modules (e-learning et classes inversées), la transversalité marquée des enseignements et une pratique

pédagogique hautement professionnalisante (développement d'applications dès les premières heures passées au Campus et apprentissage en alternance) sont au cœur du projet pédagogique.

Modalités d'évaluation et diplôme visé

La formation est sanctionnée par l'obtention du [Titre RNCP de niveau 6 \(équivalent Bac+3\) « Concepteur Développeur d'Applications »](#) du Ministère du Travail, de l'Emploi et de l'Insertion (RNCP N°31678).

Modalités d'évaluation :

- **Validation des compétences** (auto-validation et validation par un formateur professionnel de l'ensemble des compétences associées à chacun des modules de formation)
- **Réalisation de rapports mensuels relatifs à la mission professionnelle réalisée**
- **Soutenance devant un jury** accrédité par le Ministère du Travail, de l'Emploi et de l'Insertion.

Durée de la formation

La formation se décompose en deux phases :

- **9 mois de formation intensive au Campus (1204 heures) permettant d'acquérir le socle de compétences techniques et de travail en équipe ;**
- **Alternance de 12 mois comprenant 525 heures de formation au Campus afin d'élargir et d'approfondir les compétences techniques et les soft skills développées.**

Modalités d'accès et prérequis

Tous les candidats disposant d'un niveau Bac et étant demandeurs d'emploi peuvent s'inscrire à la formation, sans prérequis obligatoire. Les critères de sélection sont basés sur :

- La motivation et le projet professionnel défini ;
- La capacité à travailler en équipe et la bienveillance ;
- La curiosité, l'autonomie et la créativité.

Processus de sélection :

- Envoi d'une lettre de motivation et d'une vidéo pitch
- Evaluation avec la Méthodes de Recrutement par Simulation (MRS) de Pôle Emploi
- Immersion dans un travail collaboratif non numérique (piscine de 3 jours)

Prix de la formation

Gratuit grâce aux partenaires financeurs de la formation et les fonds de financement de la formation.

Lieux de formation

La formation DevOps est réalisée :

Campus de Grenoble

Le Totem

16 Bd Maréchal Lyautey, 38000 Grenoble

07 50 66 51 51 / audrey.graffagnino@le-campus-numerique.fr

Accessibilité aux personnes en situation de handicap

La formation est accessible aux personnes en situation de handicap, avec une adaptation possible du rythme pédagogique et des modalités d'évaluation.

Contenus Pédagogiques

1/ Socle Développeur Web&Applis – 805 heures

Module : HTML / CSS	
Durée	9 jours – 63 heures
Objectifs	<p>A l'issue de ce module vous serez capable :</p> <ul style="list-style-type: none">• Maquetter des interfaces de site internet sur papier• Créer des pages html• Mettre en forme du texte• Insérer des images• Insérer des vidéos (locales ou externes)• Ajouter des liens hypertexte• Mettre en forme des tableaux• Intégrer un formulaire de contact• Appliquer des techniques de référencement• Utiliser des feuilles de style externes• Manipuler les propriétés CSS liées au box-model• Positionner les éléments d'une page web• Gérer les mises en page responsives
Activités	<p>Les activités constitutives du module sont les suivantes :</p> <ul style="list-style-type: none">• Le mini-projet « CV » individuel• Le projet « liste de CVs » en îlot, déployé• Le projet « site d'entreprise » en îlot <p>Pour ces deux projets, réalisation de maquettes, et développement de pages web responsives (HTML5, CSS3, Bootstrap)</p>

Module : UX / UI	
Durée	6 jours – 42 heures
Objectifs	<p>A l'issue de ce module vous serez capable :</p> <ul style="list-style-type: none">• Produire ou retoucher des images via Adobe Photoshop• Concevoir et maquetter l'apparence d'un site en HTML5 et CSS3, en produire les éléments de charte nécessaires à la mise en production du projet (images, couleurs, polices de caractères etc...).
Activités	<p>Les activités constitutives du module sont les suivantes :</p> <p>Composition de visuels, retouche photo, création de charte graphique, étude d'un logotype, réalisation de mockup d'interface, mise en forme de planche de tendance.</p>

	<p>Compétences transversales travaillées sur l'ensemble des activités.</p> <ul style="list-style-type: none"> • adapter les ambitions aux délais • gérer un échange de données • travailler en équipe • défendre son travail (argumentation objective)
--	--

Module : Algorithmique	
Durée	11 jours – 77 heures
Objectifs	<p>A l'issue de ce module vous aurez acquis les concepts algorithmiques de la programmation tels que : les variables, les opérateurs, les conditions, les boucles, les tableaux.</p> <p>Vous serez capable :</p> <ul style="list-style-type: none"> • Programmer avec des variables • Programmer une condition • Programmer une fonction et l'appeler • Passer des paramètres à une fonction • Programmer une boucle • Programmer avec des tableaux
Activités	<p>Transmission des fondamentaux de l'algorithmique au travers d'activités variées : approche corporelle (je joue un algorithme), langage oral, pseudo-code, Blockly, code Master et Processing.</p> <p>Les activités constitutives du module sont les suivantes :</p> <ul style="list-style-type: none"> • Premiers pas en Algo • Projet "déménagement" • Projet "Pendule" • Projet "Jeu de l'oie" • [Optionnel] Deux projets pour aller plus loin : "Dans la tête d'un contrôleur de TER" et "Projet de manipulation de tableaux"

Module : Réseau / Système	
Durée	8 jours – 56 heures
Objectifs	<p>A l'issue de ce module vous serez capable :</p> <ul style="list-style-type: none"> • Publier un site sur internet. • Utiliser les commandes de base UNIX • Automatiser le déploiement • Maîtriser le routage • Être sensibilisé à la sécurité informatique
Activités	<p>Les activités constitutives du module sont les suivantes :</p> <ul style="list-style-type: none"> • Mini-Projet 1 : « Jeu de piste » • Projet : Publier une application sur internet • Mini-Projet 2 : Présenter votre travail • Projet : Relier le site à un nom de domaine • Projet : Faire évoluer son site, automatiser un déploiement • Mini-Projet 3 : jeu de rôle « fonctionnement internet et sécurisation » • Mini-Projet 4 : (optionnel) : Réinstaller le projet avec Docker

Module : PHP / BDD

Durée	32 jours – 224 heures
Objectifs	<p>A l'issue de ce module vous serez capable de :</p> <ul style="list-style-type: none">• Installer un environnement client/serveur WAMP• Écrire du code PHP utilisant des variables et des opérateurs, et mettant en œuvre des conditions, des boucles, des chaînes de caractères, et des tableaux• Écrire et exécuter des fonctions avec passage de paramètres• Exploiter des formulaires avec PHP• Factoriser et structurer son code• Concevoir une base de données (MySQL WorkBench)• Manipuler des données (SQL)• Accéder aux données depuis une page PHP (PDO)• Installer le framework Laravel• Utiliser le framework avec une architecture MVC (routes, controlers, templating...)• Comprendre les principes de la programmation orientée objet• Accéder aux données avec Laravel (Model)• Sécuriser une application via l'authentification• Exposer des données via une API REST
Activités	<p>Les activités constitutives du module sont les suivantes :</p> <ul style="list-style-type: none">• Projet PHP de base : « Site E-commerce »• Mini-Projet : jeu de découverte des bases de données• Projet Base de données : « Site E-commerce » :• Projet Laravel : « Site E-commerce »• Mini-Projet POO : « Warriors »• Jeu de rôle « MVC »

Module : JavaScript / WebMobile

Durée	20 jours – 140 heures
Objectifs	<p>A l'issue de ce module vous serez capable :</p> <ul style="list-style-type: none">• Développer un projet Front-End simple sans Framework utilisant les technologies HTML / CSS / JS natif• Mettre en place un projet de développement incluant le Framework Angular• Mettre en place un projet incluant le Framework ReactJS• Connaître les familles d'applications mobiles et leurs principes de développement• Connaître les principaux frameworks• Utiliser React Native• Concevoir et développer une application mobile hybride utilisant le Framework React Native• Mettre en ligne un site internet
Activités	<p>Les activités constitutives du module sont les suivantes :</p> <ul style="list-style-type: none">• Projet : « Site E-commerce »

Module : Java

Durée	18 jours – 126 heures
--------------	------------------------------

Objectifs	<p>A l'issue de ce module vous serez capable :</p> <ul style="list-style-type: none"> • Coder en utilisant la syntaxe de base Java • Implémenter les concepts de base de la programmation objet en Java (Encapsulation, Héritage, classes abstraites, Interfaces) • Manipuler des collections d'objets • Gérer les Exceptions • Accéder à des données via JDBC • Mettre en œuvre le pattern DAO • Modéliser avec UML les diagrammes de classe et de Use-cases
Activités	<p>Les activités constitutives du module sont les suivantes :</p> <ul style="list-style-type: none"> • Mini-Projet 1 : Syntaxe de base Java et concepts objets • Mini-Projet 2 : Héritage • Mini-Projet 3 : Classes abstraites et Interfaces • Mini-Projet 4 : Collections • Mini-Projet 5 : Accès aux données via JDBC • Mini-Projet 6 : Pattern DAO • Mini-Projet facultatif : découverte de Swing

Module : Soft Skills et méthodes	
Durée	11 jours – 77 heures
Activités	<ul style="list-style-type: none"> • Rentrée : 1j • Culture numérique : 1j • Communication / Théâtre : communication interpersonnelle 4j • Git : 2j • Agilité : 3j

2/ Spécialisation DevOps (399 heures)

Pédagogique par projet : chaque module est construit sur la base d'un cas d'étude à réaliser. L'étude d'un cas concret permet d'appréhender au fur et à mesure les notions / outils étudiés, sans passer par la dispense d'un cours magistral par le formateur.

Module 1 : Déploiement	
Durée	10 jours – 70 heures
Objectifs pédagogiques	<ul style="list-style-type: none"> • Réutiliser un réseau social en PHP • Déployer en local, en distant, et sur machine virtuelle • Utilisation des containers (manuel, automatique, backup, monitoring)
Activités	<p>Mettre en place le déploiement d'applications à l'aide de containers via Docker.</p> <p>Connaissances</p> <ul style="list-style-type: none"> • Mettre en place un environnement de travail • Maîtriser les concepts de base de Docker • Être capable de créer et déboguer des containers • Gérer des images localement et à distance • Savoir administrer des containers en production

	<p>Savoir-faire méthodologiques</p> <ul style="list-style-type: none"> • Installer et configurer des conteneurs Docker • Installer et configurer une machine virtuelle <p>Attitudes, posture (savoir-être)</p> <ul style="list-style-type: none"> • Autodidacte • Solliciter la communauté pour résoudre des problèmes • Apporter des solutions à la communauté d'utilisateurs • Rigueur <p>Projets réalisés</p> <p>Mise en place des mécanismes de déploiement continu à partir d'un projet réalisé antérieurement par la promotion.</p>
--	--

Module 2 : Sécurisation	
Durée	3 jours – 21 heures
Objectifs pédagogiques	Maîtrise des divers concepts de sécurité / mots de passe
Activités	<p>Sécuriser les développements de sites ou d'applications. Détecter et analyser des failles de sécurité d'une application.</p> <p>Connaissances</p> <ul style="list-style-type: none"> • Énumérer et proposer des méthodes de résolution des failles de sécurité principales dans des projets de développement • Analyser un projet existant afin de détecter les failles de sécurité • Mettre en place une politique de mots de passe dans une organisation <p>Savoir-faire technique</p> <ul style="list-style-type: none"> • Corriger les failles de sécurité classiques d'une application <p>Savoir-faire méthodologiques</p> <ul style="list-style-type: none"> • Analyser un code à la recherche de failles de sécurité • Mettre en place une méthode de conception d'application sécurisée dès la conception (Secure by Design) <p>Attitudes, posture (savoir-être)</p> <ul style="list-style-type: none"> • Rigueur • Curiosité <p>Projets réalisés</p> <p>Plusieurs projets contenant des failles de sécurité seront présentés aux apprenants, les apprenants seront tour à tour responsable sécurité et hacker. Le premier devant sécuriser l'application, et le second la pénétrer.</p>

Module 3 : Tests	
Durée	2 jours – 14 heures

Objectifs pédagogiques	Maîtriser des tests en local ou auto-unitaires
Activités	<p>Réalisation de développements pilotés par les tests (Test Driven Development). Apprentissage des spécificités de la pyramide de tests et de leur impact sur un projet de développement.</p> <p>Connaissances</p> <ul style="list-style-type: none"> • Développer une application en utilisant une logique “test first” • Différencier test unitaires / composants / intégration / bout en bout / IHM • Adapter une stratégie de tests à un code existant <p>Savoir-faire technique</p> <ul style="list-style-type: none"> • Implémentation de tests à l’aide <p>Savoir-faire méthodologiques</p> <ul style="list-style-type: none"> • Planifier le développement de tests dans un process de développement logiciel • Maîtriser les notions de développement par les tests (TDD) et développement piloté par le comportement (Behaviour Driven Development)

Module 4: Suivi des bugs	
Durée	2 jours – 14 heures
Objectifs pédagogiques	Maîtriser la définition, la capture et le tri des bugs
Activités	<p>Mettre en place un système de suivi et de gestion de bugs sous forme de tickets afin de maximiser la qualité des interactions avec les clients dans un processus Agile.</p> <p>Connaissances</p> <ul style="list-style-type: none"> • Administrer une plateforme de suivi de tickets de bugs • Configurer les droits d’accès à cette plateforme • Expliciter les différentes étapes de résolution d’un problème client <p>Savoir-faire techniques</p> <ul style="list-style-type: none"> • Installer et administrer une plateforme de suivi de tickets <p>Savoir-faire relationnels</p> <ul style="list-style-type: none"> • Expliquer aux équipes de développement l’importance de la résolution de bugs • Expliquer les enjeux d’une résolution efficace de bugs et comment communiquer sa résolution à un client <p>Attitudes, posture (savoir-être)</p> <ul style="list-style-type: none"> • Communication interpersonnelle • Pédagogie <p>Projets réalisés</p>

	Mise en place d'une plateforme de suivi de tickets pour les développeurs sur les projets de développement de la formation
--	---

Module 5: Docker	
Durée	2 jours – 14 heures
Objectifs pédagogiques	Mise en application des conteneurs avec l'outil Docker
Activités	<p>Connaissances</p> <ul style="list-style-type: none"> • Maîtriser les concepts de base de Docker • Être capable de créer et déboguer des conteneurs • Gérer des images localement et à distance • Savoir administrer des conteneurs en Production <p>Savoir-faire méthodologiques</p> <ul style="list-style-type: none"> • Installer et configurer des conteneurs Docker

Module 6: Amélioration d'une chaîne Ops	
Durée	10 jours – 70 heures
Objectifs pédagogiques	Récupérer un projet complet de DevOps et le faire évoluer.
Activités	<p>Connaissances</p> <ul style="list-style-type: none"> • Mettre en place un système de monitoring de solution Ops • Mettre en place un système de gestion des tickets • Mettre en place des tests sur l'application en cours d'utilisation pour éviter des régressions • Établir des métriques de couverture de tests sur l'application • Sécuriser la chaîne de développement • Protéger les accès avec une politique de mots de passe robuste • Tester les failles de sécurité classiques <p>Savoir-faire méthodologiques</p> <ul style="list-style-type: none"> • Installer et administrer une plateforme de suivi de tickets • Installer et administrer des solutions de télémétrie <p>Projets réalisés</p> <p>Sur un projet disponible en production, faire évoluer la chaîne Ops en se basant sur un ensemble de métriques.</p>

Module 7 : Tests UI	
Durée	2 jours – 14 heures
Objectifs pédagogiques	Mettre en place des tests d'interface utilisateur sur une application

Module 8 : Monitoring	
Durée	2 jours –14 heures

Objectifs pédagogiques	<p>Mettre en place un système de télémétrie</p> <p>Systèmes de Monitoring</p> <ul style="list-style-type: none"> • Prometheus • Redis • GraphQL • Télémétrie • Administration • Instrumentation / Log
-------------------------------	---

Module 9 : Fil rouge Obs

Durée	19.5 jours – 137 heures
Objectifs pédagogiques	Mise en place d'une stratégie globale Dev Ops sur un projet de développement en respectant toutes les bonnes pratiques abordées au cours de la formation.

Module 10 : SLA / déploiement

Durée	1 jour – 7 heures
Objectifs pédagogiques	Maîtrise des normes de qualité

Module 11 : Kubernetes

Durée	19.5 jours – 137 heures
Objectifs pédagogiques	<p>Automatiser le déploiement</p> <ul style="list-style-type: none"> • Mettre en place de déploiement de conteneurs grâce à Kubernetes • Installer et configurer Kubernetes

Module 12 : Soft Skills

Durée	2 jours – 14heures
Activités	<p>Communication interpersonnelle, codes de l'entreprise, présence numérique. Ce module est abordé de façon transverse lors de chacun des autres modules afin d'avoir une approche pratique du développement des soft skills.</p> <p>Savoir-faire:</p> <ul style="list-style-type: none"> • Présenter ses résultats • Présentation des protagonistes d'une équipe projet • S'exprimer à l'oral et à l'écrit • Réaliser une présentation Powerpoint (ou équivalent) • Reformuler une proposition client • Synthétiser une demande client • Expression orale et écrite <p>Attitudes, posture (savoir-être)</p> <ul style="list-style-type: none"> • Communication • Communication non verbale

	<ul style="list-style-type: none"> • Être attentif <p>Projets réalisés Les habiletés de présentation des résultats sont développées dans l'ensemble des projets développés dans les autres modules.</p>
--	---

Blocs de compétences travaillés en spécialisation DevOps

La formation se décompose en 12 blocs de compétences abordés, pour la majorité, de manière transverse dans les différents modules et cas d'étude travaillés.

Bloc de compétences 1 : Infra	
Objectifs pédagogiques	Maîtrise des principes, outils et mécanismes de base de Devops <ul style="list-style-type: none"> - Docker - Linux - Git (GitHub, GitLab, Git advanced) - Virtualisation - Infrastructure local ou distante (Système d'exploitation, VM, Containerization, serveur)
Bloc de compétences 2 : Tests	
Objectifs pédagogiques	Maîtriser les tests <ul style="list-style-type: none"> - Automatique (local ou distant) - Unitaires - UI / fonctionnel - Charge (Load Balance) - Couverture de tests (Test Coverage)
Bloc de compétences 3 : Bugs	
Objectifs pédagogiques	Gestion des bugs clients <ul style="list-style-type: none"> - Version - Tickets - Méthodologie - Correction - Déploiement
Bloc de compétences 4 : Sécurisation	
Objectifs pédagogiques	Sécurisation de la chaîne Ops <ul style="list-style-type: none"> - Mots de passe - Multi authentification - Infra (fire wall) - Méthodologie / pratique - Access token - Scénario d'attaque - Injection - Techniques de Hacking - RGPD - Secure by design

Bloc de compétences 5 : Déploiement	
Objectifs pédagogiques	<p>oiement et intégration continue</p> <ul style="list-style-type: none"> - Kubernetes - Load balance - Git - Cloud - Resilience / Redondance / SLA
Bloc de compétences 6 : Systèmes de sauvegarde (Backup)	
Objectifs pédagogiques	<p>Sauvegarder et restaurer de BDD</p> <ul style="list-style-type: none"> - Code - Bold - VM (snapshot) - Maîtrise de formules et calculs élaborés - Données utilisateurs
Bloc de compétences 7 : Tableaux de bord / Monitoring	
Objectifs pédagogiques	<p>Systèmes de Monitoring</p> <ul style="list-style-type: none"> - Prometheus - Redis - GraphQL - Télémétrie - Administration - Instrumentation / Log

En partenariat avec :

