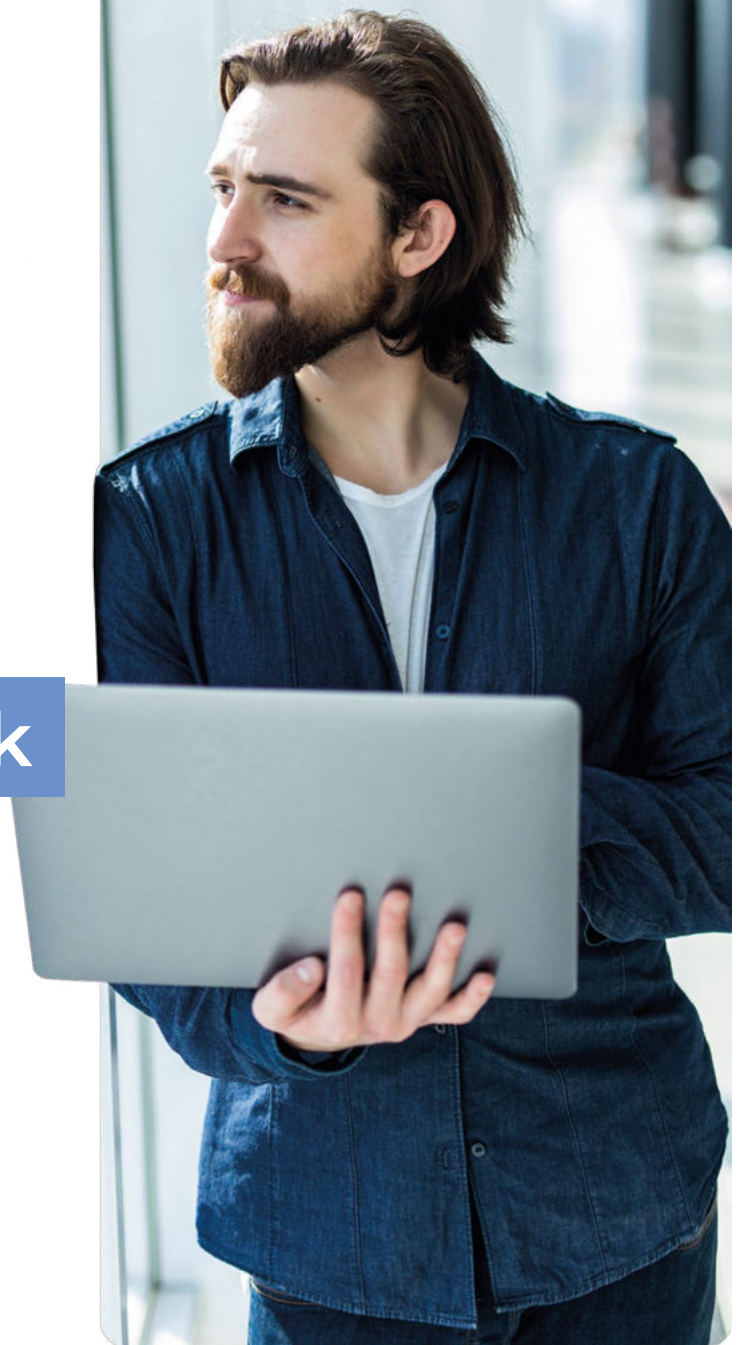




1 formation = 1 job

**Préparation
Opérationnelle
à l'Emploi de**

**Développeur·se
C# .NET Full Stack**

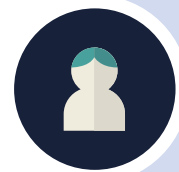


Avis Vérifiés
★★★★★ 4,6/5

Les avantages de la POEI by M2i

Sourcing et présélection des candidats

A la manière d'un cabinet spécialisé, nous mettons en place un processus de recrutement assuré par notre équipe dédiée, garantissant une sélection rigoureuse des candidats.



Suivi pédagogique

Profitez d'un suivi pédagogique régulier, assurant une transparence totale sur la progression des collaborateurs au cours de leur formation.



Certification des compétences

Valorisez les compétences de vos futurs collaborateurs grâce à l'obtention d'une certification officielle.



Formateurs experts spécialisés

Bénéficiez d'une équipe de formateurs-consultants expérimentés, spécialement sélectionnés pour assurer la qualité et la pertinence des formations dispensées dans le cadre de la POEI.



Accompagnement

Simplifiez vos démarches de recrutement en profitant d'un accompagnement personnalisé qui vous guidera à travers toutes les étapes, de l'inscription à la gestion des formalités administratives.



Partenariats stratégiques

Accédez à des contenus de formation de qualité grâce à nos partenariats stratégiques avec les plus grands éditeurs.



Conseil en GEPP

Optimisez votre stratégie GEPP (Gestion des Emplois et des Parcours Professionnels) en combinant la POEI avec d'autres dispositifs proposés par le groupe M2i, tels que l'alternance, pour créer un parcours adapté aux besoins spécifiques de votre entreprise.



Développeur·se C# .NET Full Stack

La plupart de nos cursus sont réalisables en formation distancielle.

OBJECTIFS DE FORMATION

Ce cursus permet aux apprenants de monter en compétence sur le développement d'applications Web avec la technologie C# .NET Core, HTML/CSS/JavaScript et Angular et déployer ces applications sur la plateforme Azure.

LES MISSIONS DE DÉVELOPPEUR·SE

- Analyser et respecter le cahier des charges établi par le·la chef·fe de projet
- Définir les différentes étapes de fonctionnement de l'application
- Etudier l'ensemble des solutions techniques possibles
- Développer les applications en respectant les contraintes établies
- Tester et mettre à jour des applications
- Produire une documentation technique et fonctionnelle des applications

MÉTIERS CIBLES

Développeur·se .NET, Développeur·se C#, Développeur·se Full Stack, Développeur·se Web.

PRÉREQUIS

Bac+3/+4 informatique ou scientifique ou expérience équivalente - Capacité d'analyse et de synthèse - Rigueur et sens de la méthode - Connaissance de l'anglais et d'un langage de programmation est un plus.

PUBLIC CONCERNÉ

Demandeur·se d'emploi inscrit·e à France Travail.

MODALITÉS, MOYENS ET MÉTHODES PÉDAGOGIQUES

Formation dispensée en présentiel et/ou à distance par des formateurs professionnels : apports théoriques, exercices de mise en situation professionnelle.

Les formateurs alternent entre la méthode affirmative (présentations, démonstrations), interrogative (tests, quiz...) et active (jeu de rôles, cas pratique, apprentissage par les pairs ...). Ils s'appuient sur les 4 piliers de l'apprentissage issus des dernières découvertes en neurosciences. Les journées sont rythmées par l'alternance entre théorie et pratique via un jeu d'exercices : exercices progressifs, exercices récapitulatifs et de validation des acquis.

ADMISSION

- Dossier de candidature
- Test qui permettra ensuite de préconiser d'éventuelles solutions de remise à niveau
- Entretien

MODALITÉS D'ACCÈS

Le groupe M2i s'engage pour faciliter l'accessibilité de ses formations. Les détails de l'accueil des personnes en situation en handicap sont consultables sur la page [Accueil et Handicap](#).

VALIDATION DES ACQUIS

- Dans chaque module, environ 50% de travaux pratiques pour valider la compréhension
- Tout au long du parcours, 9,5 jours de travaux pratiques sur le projet « fil rouge » en semi-autonomie pour ancrer les acquis
- En fin de parcours, 1 jour pour la préparation et le passage du test de connaissance M2i POE-DOTNET

DÉLAIS D'ACCÈS

La candidature peut être déposée jusqu'à 15 jours avant la date de démarrage de la session.

COMPÉTENCES ET TECHNOLOGIES ABORDÉES

57 jours - 399 heures

Le ratio cours théorique/travaux dirigés est de 50%.

Catégorie	Modules	Durée (Jours)	Durée (Heures)
Présentation	Présentation du cursus, des plateformes pédagogiques et du projet final « fil rouge ».	0,5	3,5
Méthodes	Algorithmique et programmation structurée : Enumérer les connaissances nécessaires à l'apprentissage d'un langage de développement - Reconnaître les structures de base de la programmation (boucles, conditions) - Identifier les grands paradigmes de programmation (procédural, objet) - Distinguer la notion d'objet et les concepts associés - Disposer d'un premier point de vue avec C# - Identifier les variables et le typage des données.	1,5	10,5
	Fondamentaux du Web : Client Serveur et leur relation. Les API, Les SGBD et bases de données, les protocoles du Web, Serveur DNS...	1	7
Versionning et test	Git/Github/Gitlab : Décrire les principes d'un gestionnaire de versions distribué - Identifier par la pratique, la philosophie de Git et ses apports - Plateforme de collaboration GitHub et Gitlab.	2	14
Méthodes	Méthodes agiles : Comprendre la démarche - Scrum et Jira par la pratique.	1,5	10,5
Validation des acquis	Travaux pratiques.	0,5	3,5
Méthodes	Introduction à la programmation orientée objet et modélisation UML : Le paradigme de POO, class, objets, héritage, encapsulation, polymorphisme, bonnes pratiques, maintenabilité - Définition d'UML - Les diagrammes de cas d'utilisation, de class et de séquences.	1	7
	Architecture des applications .NET : Connaître les architectures typiques du monde .NET - Savoir sur quels critères juger une architecture technique - Sélectionner les technologies .NET adéquates - Comprendre comment communiquer avec d'autres technologies - Choisir la bonne méthodologie de développement - Structurer correctement une application dans Visual Studio - Choisir et définir une convention de nommage dans son architecture.	1	7
Développement back	SQL et Transact-SQL : Utiliser la méthodologie Merise - Rappeler les principaux concepts des SGBDR (Système de Gestion des Bases de Données Relationnelles) et d'algèbre relationnelle utilisés dans le langage SQL - Interroger une base de données avec la clause SQL SELECT - Utiliser les commandes SQL de mise à jour des données - Connaître les commandes SQL de début et fin de transaction BEGIN, COMMIT et ROLLBACK - Appréhender les concepts de gestion des privilèges systèmes et objets avec les commandes SQL GRANT et REVOKE - Créer, modifier et supprimer certaines catégories d'objets (table, index, vues...) avec CREATE, ALTER et DROP.	3	21
	C# - Développement avec le Framework .NET Core 3 : Présentation de .NET Core - Le langage C# - L'héritage - Les fonctionnalités avancées - LINQ - API .NET Core - TPL - Manipulation de données XML et JSON - Utilisation des Web Services.	5	35
	Introduction à WPF : Créer une interface basique WinForms - Concevoir une interface graphique avec WPF - Utiliser le Two Way Data Binding - Exploiter les possibilités du langage XAML - Introduction à MVVM.	2	14
Développement front	Les fondamentaux d'HTML et CSS : Décrire la structuration d'une page HTML 5 - Ajouter des styles CSS aux éléments d'une page - Utiliser les blocs et les tableaux - Créer des formulaires - Mettre en œuvre des pages HTML.	2	14
Versionning et test	Tester son code : Comprendre les principes de développement des tests - Ecrire et utiliser un test unitaire en C# - Utilisation de MSTest et NUnit.	1	7
Validation des acquis	Travaux pratiques fil rouge.	2	14
Développement back	C# - Entity Framework Core 3.0 : Identifier les mécanismes avancés de Entity Framework Core 3 et ses fonctionnalités - Accéder aux données avec ADO.NET - Définir l'Object Relational Mapping (ORM) - Utiliser LINQ to Entities - Gérer le mapping avancé - Décrire les performances d'exécution des requêtes - Migrer le schéma de la base de données.	4	28
Validation des acquis	Travaux pratiques - Développement du projet « fil rouge ».	2	14

Catégorie	Modules	Durée (Jours)	Durée (Heures)
DevOps	DevOps : Comprendre l'usage des conteneurs avec Docker et mettre en place un exemple concret - Introduction à l'intégration continue avec Jenkins ou GitLab.	1	7
Développement front	JavaScript : Connaître les bases de JavaScript et de son utilisation pour le DOM - Gérer les événements et les manipulations dynamiques - Connaître les règles d'or de la programmation avec JavaScript.	3	21
Développement back	Développer des applications Web ASP.NET MVC Core 3 : Développer une application Web en utilisant ASP.NET - Utiliser des fonctionnalités JavaScript / jQuery en interaction avec ASP.NET - Créer des Web Services RESTful (HTTP et JSON) - Déployer une application ASP.NET.	5	35
Validation des acquis	Travaux Pratiques - ASP.Net Core - « fil rouge ».	2	14
Développement back	Développer des services Web REST en C# : Introduction aux services Web - Implémentation en C# - Déploiement et cycle de vie d'un service - Cross Origin Resource Sharing - Sécuriser un service Web.	2	14
Développement front	Introduction à ES6/TypeScript : Les apports d'Angular - Comprendre la structure du Framework - Composants, template et liaisons de données - Directives et pipes - Services et injection de dépendances - Formulaires - RxJS - Routing et navigation - Utiliser Angular avec ASP.NET.	1	7
	Angular pour .NET : Décrire les principes d'un gestionnaire de versions distribué - Identifier par la pratique, la philosophie de Git et ses apports - Plateforme de collaboration GitHub et Gitlab.	4	28
Développement Cloud	Microsoft Azure - Développer une application Cloud en C# : Prendre en main le SDK Azure dans Visual Studio dans le cadre de Azure Compute - Manipuler les différents types de stockage avec Azure Storage - Gérer la base de données avec SQL Azure Database (création, administration et consommation) - Sécuriser les applications et les api web avec Azure Active Directory - Réaliser des backend mobile avec Azure Mobile Services.	4	28
Validation des acquis	Finalisation du projet « fil rouge » dans Azure.	4	28
	Préparation et passage du test de connaissance M2i POE-DOTNET	1	7

57 jours 399 heures

Programme et planning détaillé sur demande

LES PLUS DE M2i FORMATION

Microsoft Teams Education, un outil de suivi et d'animation en présentiel et à distance

- Espace de stockage collaboratif pour accéder aux différents supports de cours et cahiers d'exercices.
- Espace dédié aux formateurs pour les fiches de suivi des apprenants qui permettent d'indiquer le niveau d'acquisition des connaissances de chaque apprenant.
- Outil de gestion des TPs pour suivre le niveau de progression.
- Intégration de la plateforme WooClap pour proposer des activités d'apprentissage interactives.
- Fonctionnalités pour gérer des sessions à distance.

La playlist e-Learning*

Tous les apprenants ont accès avant, pendant et après le cursus à notre plateforme e-learning ACADEMIIC pour :

- Acquérir les connaissances prérequis avant de démarrer (sous condition de l'existence des modules pour combler les lacunes).
- Utiliser les modules conseillés par les formateurs pour faire de l'ancrage mémoriel sur des sujets abordés pendant la formation.
- Revenir sur un sujet après la formation pour continuer à s'auto-former.

* pour ce cursus, certains modules e-learning sont en anglais

Je souhaite recruter des Développeur·se·s C# .NET Full Stack



Les étapes de la POEI by M2i

